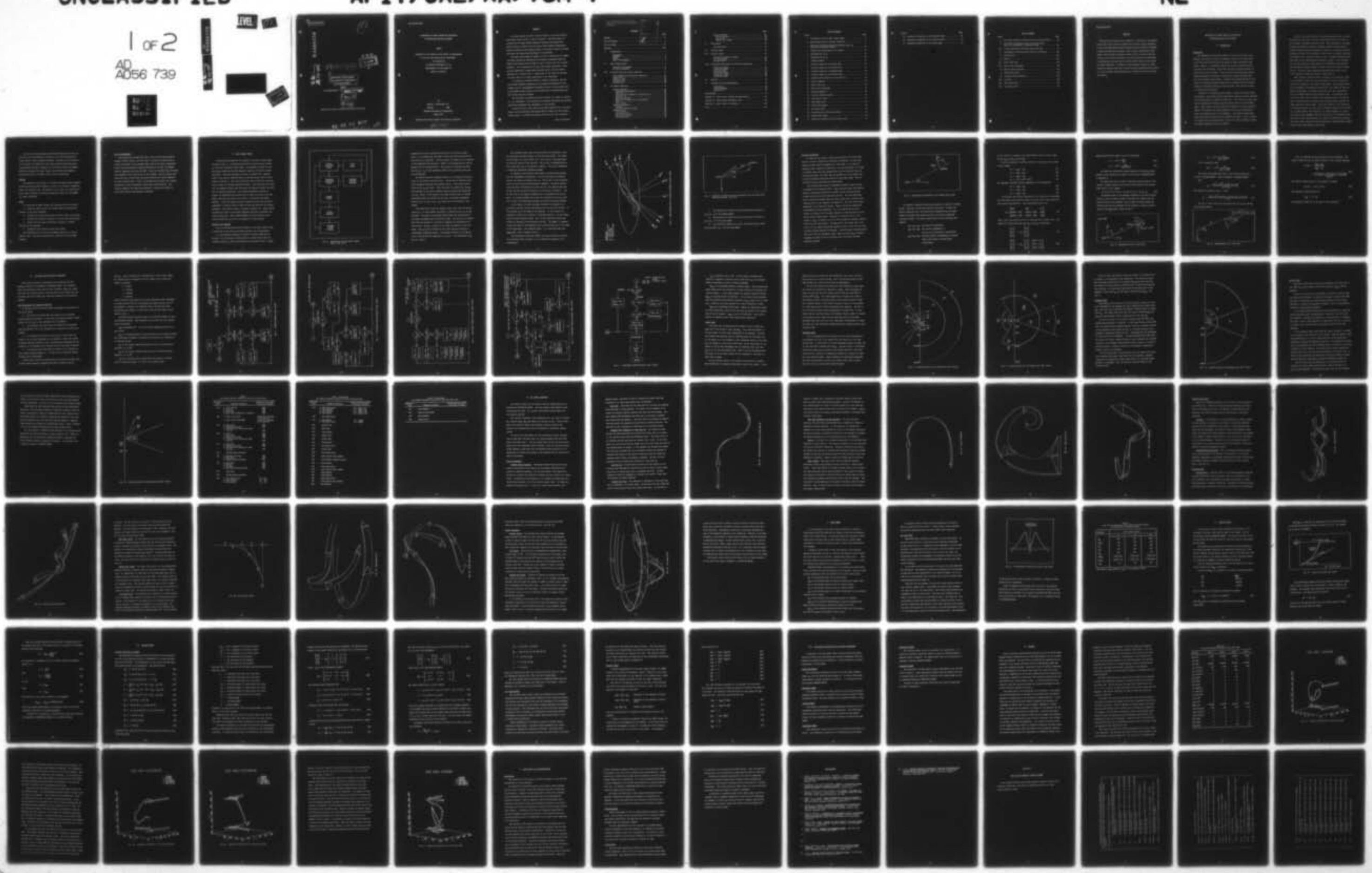
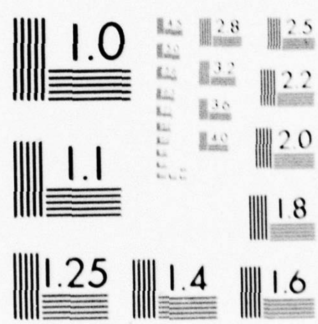
AD-A056 739 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO SCH--ETC F/G 15/7
DEVELOPMENT OF SMART TARGET FOR SIMULATION OF ONE-ON-ONE AIR-TO--ETC(U)
MAR 78 D A LEUTHAUSER

UNCLASSIFIED AFIT/GAE/AA/78M-7 NL

1 OF 2
AD
A056 739





MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF STANDARDS-1963-A

(14) AFIT/GAE/AA/78M-7

AD A056739

AD No.
DDC FILE COPY

DDC
RECEIVED
JUL 19 1978
F

(9) Master's thesis,

(6) DEVELOPMENT OF SMART TARGET FOR SIMULATION OF ONE-ON-ONE AIR-TO-AIR COMBAT.

THESIS

AFIT/GAE/AA/78M-7 (10) Dennis A. Leuthauser
~~Captain~~ USAF

(11) Mar 78

(12) 166p.

Approved for public release; distribution unlimited

78 07 07 008

012225

AFIT/GAE/AA/78M-7

DEVELOPMENT OF SMART TARGET FOR SIMULATION
OF ONE-ON-ONE AIR-TO-AIR COMBAT

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

by

Dennis A. Leuthauser, B.S.

Captain USAF

Graduate Aeronautical Engineering

March 1978

Ø12225

Preface

Air-to-air combat has been of great interest to me since being an operational fighter pilot in the F-4 aircraft. When the subject of working with the Smart Target Computer Simulation was proposed by Captain Philip L. Abold of the Air Force Flight Dynamics Laboratory/ Systems Integration and Analysis Branch, I was eager to apply my flying experience and engineering knowledge to this problem.

The Smart Target Model is unique, in that it solves the air combat simulation problem by duplicating, as closely as possible, what the human pilot observes and the logic he uses to select and fly a certain maneuver. As an operational F-4 pilot with combat experience, I have had the opportunity to make these decisions and fly the air combat maneuvers on a regular basis. A great deal of the work that has gone into this thesis is a direct application of my experience.

I am indebted to Captain Philip L. Abold, my sponsor at the Air Force Flight Dynamics Laboratory, for providing the proposal of the problem, and his encouragement throughout my work on this project, and to Lieutenant Richard Floyd for his initial help in understanding the Smart Target Computer Program.

The assistance provided by my thesis advisor, Dr. Robert A. Calico, Jr., was invaluable. His direction and continued motivation were greatly appreciated throughout the development of this project.

And most of all, I must express my deepest appreciation to my wife, Linda, for her continuing love and patience in support of this thesis, without which it is likely the project would not have been completed.

Dennis Leuthauser

Per 7-25-78 telecon with AFIT (Mr. Wolaver) Please delete the classified references

Contents

List of Figures

Figure		Page

List of Tables

Abstract

Research and evaluation were conducted to determine if an adaptive maneuvering target could be utilized for air-to-air combat simulation in the Large Amplitude Multimode Aerospace Research Simulator (LAMARS). A computer program was developed, based on Quest Corporation's Smart Target, which enabled the target to make tactical decisions based on aircraft states, make variations in decision parameters corresponding to differences in pilots, and to provide control inputs to fly actual air combat maneuvers. Validation of the simulated target was accomplished by numerous test runs to ensure simulated maneuvers were realistic, and continuity between maneuvers at different pilot skill levels was valid.

DEVELOPMENT OF SMART TARGET FOR SIMULATION

OF ONE-ON-ONE AIR-TO-AIR COMBAT

I. Introduction

Background

The simulation of air-to-air combat is not a new field (Ref 1:1). However, the greatest strides in developing realistic air combat simulators have been achieved since the advent of digital and hybrid computers. Many U. S. Government agencies and private corporations have developed such simulators. There have been many techniques employed in simulating air-to-air combat. These include differential games theory, numerical methods in trajectory optimization, energy maneuverability, and adaptive maneuvering targets. Although these approaches may be very useful in determining optimal flight paths, evaluating missile guidance systems, or designing automatic control systems, they lack the realism of a human piloted aircraft.

Therefore, in simulating aerial combat, a realistic piloted target is desirable. Many other adaptive maneuvering targets have been developed with varying degree of complexity (Refs 2-5). The Tactics II target uses one all encompassing tactical rule and limits itself to the horizontal plane. Other models, such as COAP and TAC Avenger, are able to generate some of the classical air combat maneuvers, but are too deterministic within the defensive part of an engagement. The Adaptive Maneuvering Logic (AML) model provides the best maneuvering target to date. However, the target has proved unrealistic, in that it is able to maneuver beyond the normal capabilities of most aircraft.

In 1976, the Aerospace Medical Research Laboratory (AMRL) at Wright-Patterson AFB, Ohio saw a need for a maneuvering target to be used in their Dynamic Environment Simulator (DES). They contracted to the Quest Research Corporation to provide a computer simulation which would incorporate target logic into equations of motion of a simulated aircraft, determine the state of the aircraft in real time, and display the target continuously over a subject's field of view. The Quest Corporation surveyed many of the above mentioned simulation systems. They concluded that the best overall tactical model for a responsive target was the one developed by Captain Robert P. Barrett in his AFIT Master's Thesis (Ref 6). The Smart Target computer program was then written by the Quest Corporation based upon Barrett's model (Ref 7).

Smart Target is unique in that it responds to the maneuvers flown by the simulator pilot, which in turn forces the pilot to change his controls in response to the target's maneuvering. But, because the target is intended to fly against a human pilot, this target has been designed to duplicate, as closely as possible, the logic utilized by a pilot in reaching decisions and executing them. To accomplish this, the entire combat space is divided into situation cells. For each of these situation cells, air combat tactics dictate which classical air combat maneuvers will be flown. This model contains a full repertoire of tactical maneuvers, so that it may effectively operate in all regions of the combat space. The selection of a certain maneuver in a given situation is also dependent upon the skill level of the simulated pilot. The pilot's skill level and hence his decision process may be altered. This skill level ranges from the over-aggressive, over-confident fighter pilot, to the newly trained, timid pilot.

The Air Force Flight Dynamics Laboratory/Systems Integration and Analysis Branch (AFFDL/FGD) is involved in an F-106 Integrated Fire Flight Control (IFFC) simulation program. One phase of this program is to investigate the adequacy of their Large Amplitude Multimode Aerospace Research Simulator (LAMARS) in an air-to-air combat environment. AFFDL/FGD concluded that Smart Target would provide the necessary maneuvering target to validate their air-to-air simulation.

Problem

The problem of this thesis is to evaluate the Smart Target computer program and make necessary changes, so that it may provide a responsive target for simulator use. The purpose of this study is to determine if Smart Target provides an acceptable maneuvering target for the LAMARS air combat simulation.

Scope

In evaluating the Smart Target, the following areas are covered:

1. Check the logic by which the target selects an appropriate maneuver in any given situation.

2. Evaluate the factors upon which the target bases its decision.

3. Redefine the controls necessary to fly each selected maneuver.

4. Ensure that the target continuously responds to changes initiated by the opponent.

5. Experiment with different pilot skill levels.

This evaluation was not run on the LAMARS system due to lack of computer time. The entire simulation was conducted on the CDC 6600 computer.

Plan of Development

The program was obtained from AMRL. Since they had made numerous changes to Smart Target, the first step in solving this problem was to return the program to the original format used by the Quest Corporation. Second, various subroutines required modification so that the program would be compatible with the CDC 6600. Third, an aircraft model developed using dynamics of an F-4E/J aircraft was attached. Fourth, various test cases were run to verify that the target selected the correct maneuver for each situation cell. Fifth, the desired control inputs were altered extensively so that the maneuver flight paths were realistic. And finally, numerous test cases were executed to evaluate the effect of pilot skill level on the maneuvers simulated.

II. Smart Target Theory

A generalized information flow diagram of the Smart Target model is shown in Fig. 1. It shows how the tactics selection logic relates to the entire responsive target model. The opponent aircraft and target dynamic models provide the geometric inputs to the tactics selection model. These are discussed in detail in Chapter VII. Another input to the tactics selection logic is the pilot model, which is discussed in Chapter V. The tactics selection logic outputs the situation cell in which the aircraft presently resides. This situation cell along with the dynamical inputs from the target and opponent provide the basis for the maneuver generation. The maneuver generation, covered in Chapter IV, determines the desired value of angle of attack, side slip angle, bank angle and thrust to be used in a particular air combat maneuver. Since the desired controls frequently exceed the aircraft's capability, a control filter is used to limit the values of the control variables. This control filter is covered in Chapter VI. These control variables are input to the target dynamical equations to generate the actual flight path. These target dynamics are then fed back to the tactics selection logic and to the opponent's visual display.

Aircraft Axis Systems

Prior to developing the actual dynamics of the Smart Target model, a discussion of the various coordinate systems used is appropriate.

In every dynamics problem, an inertial reference frame must be defined. For the purpose of this thesis, an earth surface fixed coordinate system, F_E, will be considered as the inertial frame. Several

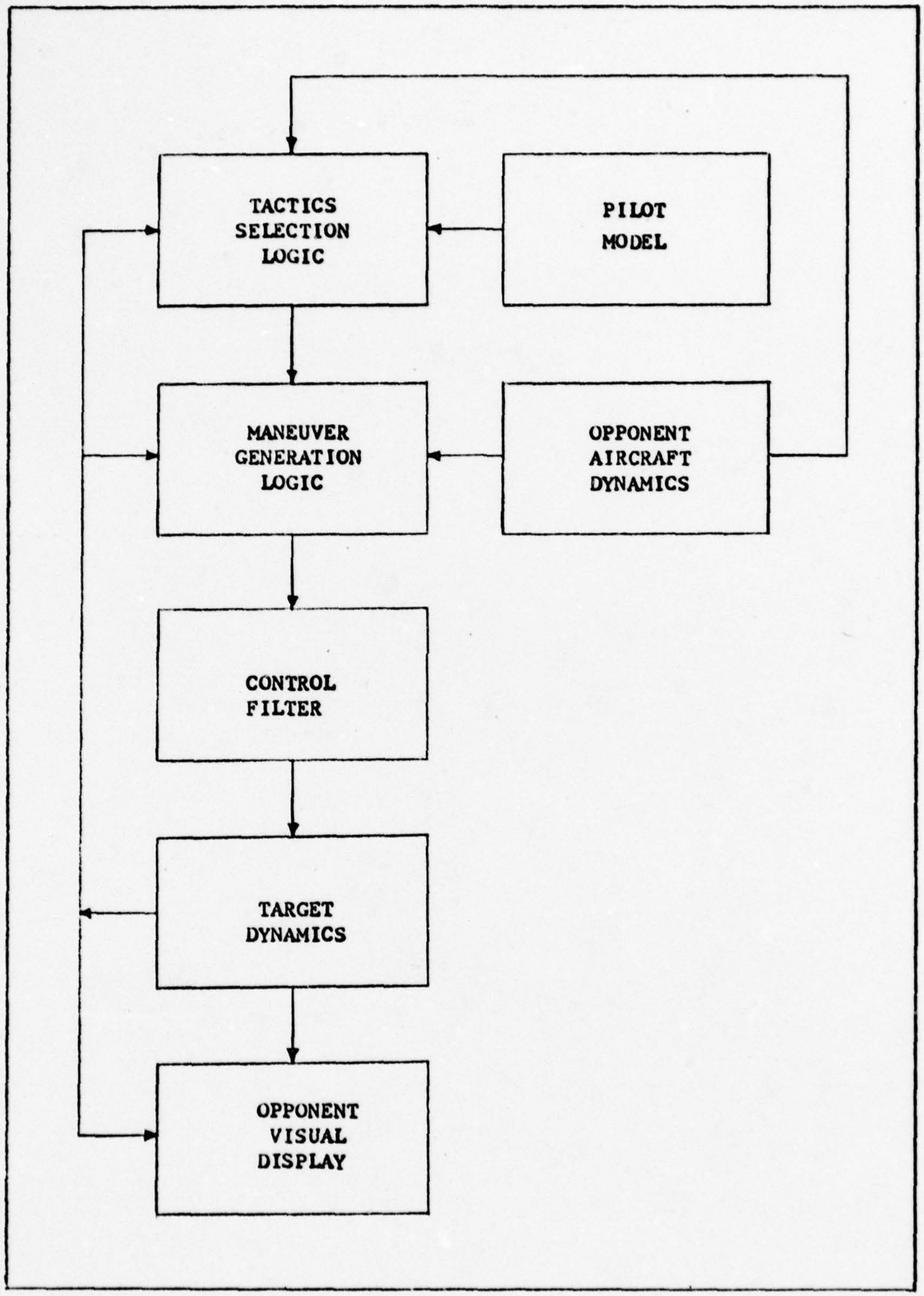


Fig. 1. Information Flow for Smart Target Model (Ref 7:3-3)

assumptions are made regarding the choice of this inertial frame. First, it is assumed that the earth is flat, and that the rotation of the earth can be neglected. Second, gravity is assumed to be a constant in both magnitude and direction. Third, it is assumed that no wind or movement of the air mass exists. The z_E axis is directed vertically down and $x_E - y_E$ is the horizontal plane, with x_E pointing north and y_E pointing east.

The vehicle vertical reference frame, F_V, is attached to the aircraft at the center of gravity (c.g.). The z_V axis is directed vertically downward, along the local gravity vector. The x_V axis points north and the y_V axis east. Since the earth is assumed flat, F_V has axes parallel to F_E, and the angular velocity between the frames, $\bar{\omega}^V$, is zero. Hence, if the origin of the earth surface frame is located immediately below the vehicle at time zero, the target coordinates would be $x_E = 0$, $y_E = 0$, $z_E = -h_T$, where h_T is the altitude of the target.

The target pilot views his opponent from a body fixed coordinate system, F_B. In this system, the origin is located at the c.g. of the aircraft. The x_B axis is directed forward along the intersection of the plane of symmetry and the waterline plane. The z_B axis is oriented down from the aircraft in the plane of symmetry and perpendicular to the x_B axis. The $x_B - z_B$ plane is the plane of symmetry of the aircraft. The y_B axis is oriented out the right wing and completes a right-handed orthogonal system. The angular velocity of F_B relative to F_E is $\bar{\omega}$, and has the components p, q, and r. The components of $\bar{V}_B$ are u, v, and w.

The reference frame used for the aircraft force equations is the air trajectory reference frame, or wind axes (Ref 8:109). The origin of this frame is at the aircraft c.g. The x_w axis is directed along the velocity vector of the aircraft. The z_w axis is perpendicular to the x_w axis and lies in the plane of symmetry. The y_w axis completes a right-hand orthogonal coordinate system.

In addition, the stability axis system is utilized when dealing with the aerodynamic coefficients. In this right-handed coordinate system, the origin is fixed at the c.g. The x_s axis is the projection of the x_w axis on the aircraft plane of symmetry. The angle between these two axes is defined as the side slip, β. The angle between the x_s axis and the x_B axis is defined to be the angle of attack, α. The z_s axis lies in the plane of symmetry and is coincident with the z_w axis. In a similar manner the y_s axis is coincident with the y_B axis. Fig. 2 shows the relationship between the various coordinate systems.

In air combat, the pilot actually considers his opponent in a spherical coordinate system superimposed upon the body axis system. In this system, R_F is the range to the opponent. In this thesis, the subscript T refers to the target or reference aircraft or to the target fixed vertical reference frame. The F or OPP subscripts denote the fighter or opponent or non-reference aircraft. The symbol ζ_F denotes the azimuth angle to the opponent, and is measured positive to the right in the $x_B y_B$ plane. The elevation angle, η_F, is positive above the $x_B y_B$ plane. This is shown in Fig. 3.

The orientation of F_B with respect to F_V may be found by consecutive rotations about the axes z, y, x, through the angles ψ, θ, ϕ, respectively.

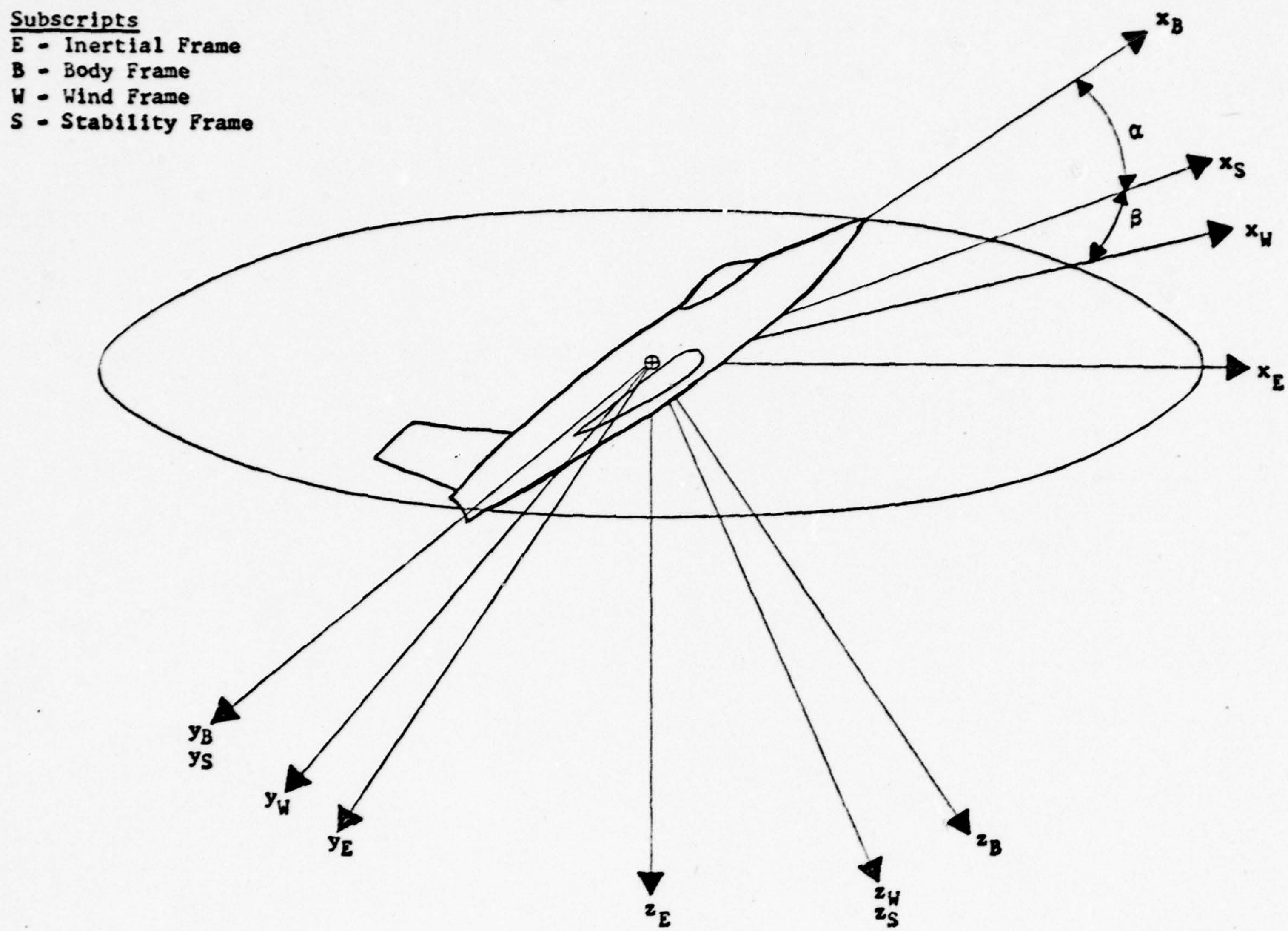


Fig. 2. Axis Systems Used in Aircraft Simulation (Ref 7:3-24)

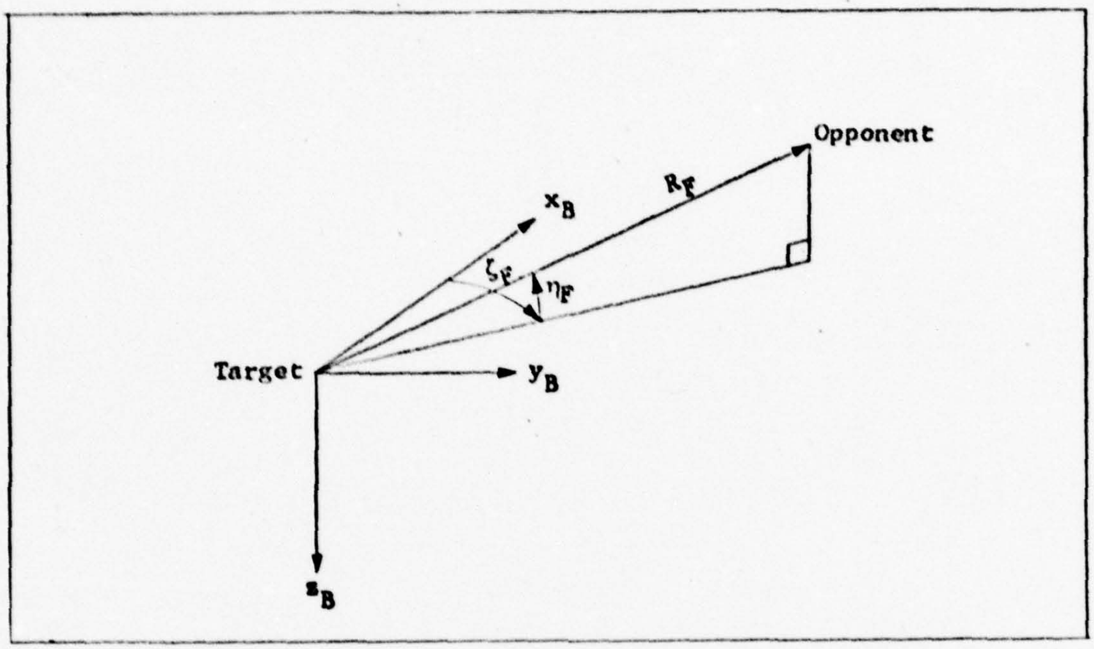


Fig. 3. Spherical Coordinate System Superimposed Upon the Body Fixed Coordinate System (Ref 6:9)

1. The rotation ψ is about the z_V axis and carries the axes to x_2 y_2 z_2. ψ is the heading angle.

2. The rotation θ is about the y_2 axis and carries the axes to x_3 y_3 z_3. θ is the pitch angle.

3. The rotation ϕ, about the x_3 axis, carries the axes to their final position, F_B. ϕ is the bank angle.

Decision Parameters

In order for the target to react realistically, the actual logic used by a pilot in reaching his decision is simulated. At very long ranges the pilot sees his opponent as a point mass. He is able to estimate range, azimuth, and elevation. As time progresses, he is able to estimate range rate, $\dot{R}_F$, azimuth rate, $\dot{\zeta}_F$, and elevation rate, $\dot{\eta}_F$. As range decreases, the pilot sees the opponent as a rigid body, with a specific angular orientation within his body-fixed frame. This is the level at which decisions are usually made.

The situation space used in developing the Smart Target tactics is composed of 37 situation cells. These situation cells were defined because a meaningful difference in combat tactics selection exists between each cell and the surrounding regions. The five dimensions used to define the combat space are range, range rate, steering error, steering error of the opponent, and angle off. The pilot views these parameters in his body axis system. Figure 4 shows the geometrical relationship of these dimensions. θ_T, the steering error of the target, is defined to be the angle between the velocity vector of the target and the range vector. Likewise, the steering error of the opponent, θ_{OPP}, is defined to be the angle between the opponent velocity vector and the range vector. The angle off of the target, ϕ_T, is defined to be the angle between the opponent velocity vector and the line-of-sight extended through the opponent. Situation space dichotomization logic uses the five variables, range, range rate, θ_T, θ_{OPP}, and ϕ_T to determine in which of the 37 situation cells the target aircraft presently resides.

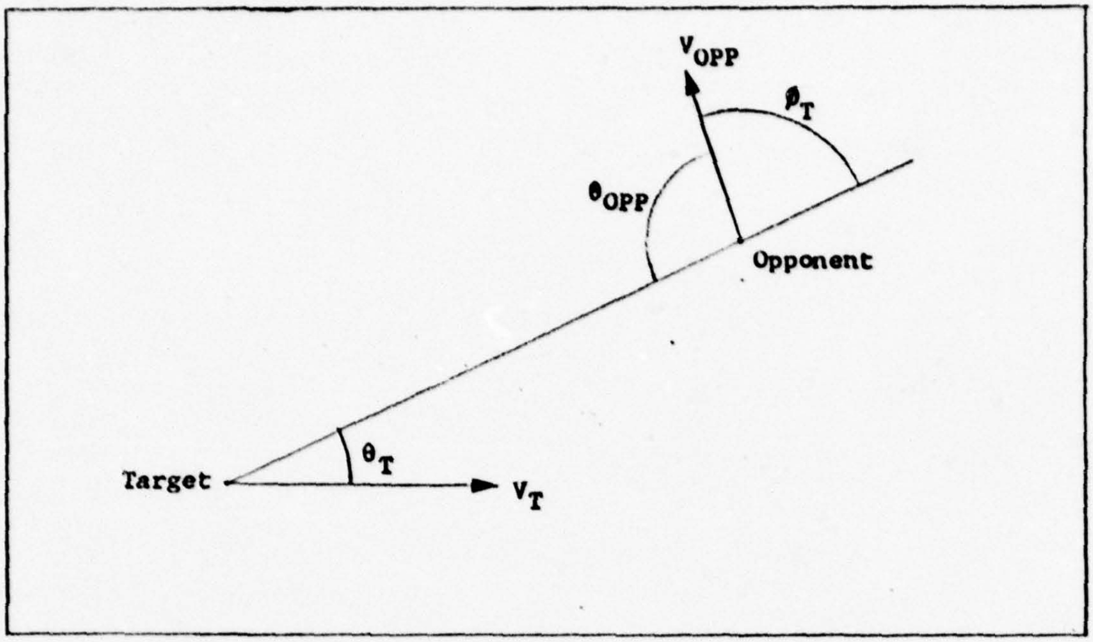


Fig. 4. Geometrical Relationship in Air Combat (Ref 7:3-5)

The computer simulation continuously updates the target's relative states. These are in the earth surface fixed frame. To enter the tactics selection logic, the earth-fixed coordinates must be transformed to the body axis system and then to the pilot's spherical coordinate system. The target states consist of the following components:

x_{TE}, y_{TE}, z_{TE},	the position coordinates.
u_{TE}, v_{TE}, w_{TE},	the velocity components in the x_E, y_E, z_E directions, respectively.
ψ_{TE}, θ_{TE}, ϕ_{TE},	the Euler angles corresponding to heading angle, pitch angle, and bank angle, respectively.

For the fighter or opponent, the corresponding terms are labeled x_{OPP} through w_{OPP} and ψ_{FE} through ϕ_{FE}.

First the opponent's relative position is determined in the target vertical frame:

$$\Delta x = x_{OPP} - x_{TE} \tag{1}$$

$$\Delta y = y_{OPP} - y_{TE} \tag{2}$$

$$\Delta z = z_{OPP} - z_{TE} \tag{3}$$

$$R_F = (\Delta x^2 + \Delta y^2 + \Delta z^2)^{\frac{1}{2}} \tag{4}$$

The opponent's relative velocity components in this frame are:

$$\Delta u = u_{OPP} - u_{TE} \tag{5}$$

$$\Delta v = v_{OPP} - v_{TE} \tag{6}$$

$$\Delta w = w_{OPP} - w_{TE} \tag{7}$$

The position and velocity components of the fighter are transformed from the target fixed vertical frame to the body fixed frame through an Euler angle transformation. The transformation matrix is:

$$L_{BV} = \begin{bmatrix} c\theta c\psi & c\theta s\psi & -s\theta \\ s\phi s\phi c\psi - c\phi s\psi & s\phi s\theta s\psi + c\phi c\psi & s\phi c\theta \\ c\phi s\theta c\psi + s\phi s\psi & c\phi s\theta s\psi - s\phi c\psi & c\phi c\theta \end{bmatrix} \tag{8}$$

where c and s are abbreviations for cosine and sine, respectively. The angles ϕ, θ, and ψ represent ϕ_{TE}, θ_{TE}, and ψ_{TE}, respectively.

$$\begin{bmatrix} x_F \\ y_F \\ z_F \end{bmatrix} = L_{BV} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} \tag{9}$$

$$\begin{bmatrix} u_F \\ v_F \\ w_F \end{bmatrix} = L_{BV} \begin{bmatrix} \Delta u \\ \Delta v \\ \Delta w \end{bmatrix} - \begin{bmatrix} qz_F - ry_F \\ rx_F - pz_F \\ py_F - qx_F \end{bmatrix} \tag{10}$$

Azimuth and elevation angles can now be determined.

$$\eta_F = \sin^{-1}\left(\frac{-z_F}{R_F}\right) \tag{11}$$

$$\zeta_F = \sin^{-1}\left(\frac{y_F}{R_F \cos \eta_F}\right) \tag{12}$$

The rules for selecting the proper quadrant for these and other angles in this chapter may be found in the listing of PROGRAM SMTTGT in Appendix A.

The range rate, $\dot{R}_F$, is found by taking the scalar product of the fighter's velocity vector in the target's body frame and the line-of-sight vector to the fighter.

$$\dot{R}_F = u_F \cos \eta_F \cos \zeta_F + v_F \cos \eta_F \sin \zeta_F - w_F \sin \eta_F \tag{13}$$

The heading angle of the opponent expressed in the body centered frame, ψ_F, must be found in order to determine the azimuth rate and elevation rate of the opponent. ψ_F is simply the angle between the target x_B axis and the component of the opponent's relative velocity vector in the $x_B y_B$ plane. Refer to Fig. 5.

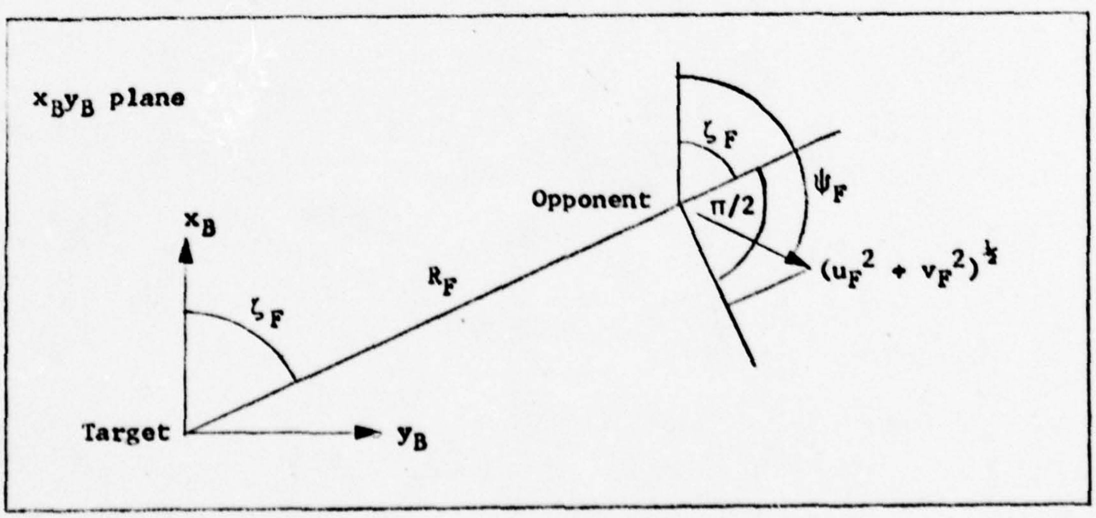


Fig. 5. Determination of $\dot{\zeta}_F$ (Ref 6:16)

$$\psi_F = \cos^{-1} \frac{u_F}{(u_F^2 + v_F^2)^{\frac{1}{2}}} \tag{14}$$

If v_F is negative, then

$$\psi_F = -\cos^{-1} \frac{u_F}{(u_F^2 + v_F^2)^{\frac{1}{2}}} \tag{15}$$

The special cases where Eqs (14) or (15) become singular are treated in PROGRAM SMTTGT. Refer to Appendix A, PROGRAM SMTTGT.

Now,

$$\dot{\zeta}_F = \frac{(u_F^2 + v_F^2)^{\frac{1}{2}} \cos\left(\frac{\pi}{2} + \zeta_F - \psi_F\right)}{R_F \cos \eta_F} \tag{16}$$

This equation is shown in Fig. 5. Also,

$$\dot{\eta}_F = \frac{(u_F^2 + v_F^2)^{\frac{1}{2}} \cos(\psi_F - \zeta_F) \sin \eta_F - w_F \cos \eta_F}{R_F} \tag{17}$$

See Fig. 6, which shows the plane containing the z_B axis and the line-of-sight vector.

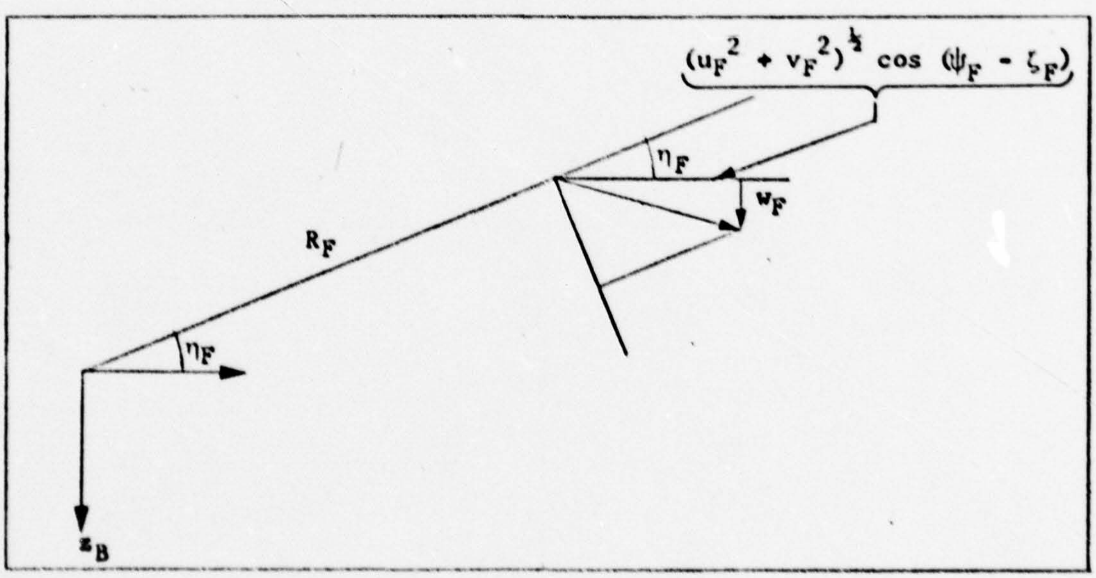


Fig. 6. Determination of $\dot{\eta}_F$ (Ref 6:17)

Next, the steering errors and angles off are determined. The target's angle off, ϕ_T, is found according to the following equation:

$$\cos \phi_T = \frac{\overline{V}_{OPP} \cdot \overline{R}_F}{\overline{V}_{OPP} \quad \overline{R}_F}$$

$$= \frac{(\Delta x)(u_{OPP}) + (\Delta y)(v_{OPP}) + (\Delta z)(w_{OPP})}{V_{OPP}\ R_F} \tag{18}$$

The target's steering error to the opponent is simply:

$$\cos \theta_T = \cos \zeta_F \cos \eta_F \tag{19}$$

The opponent's steering error is:

$$\theta_{OPP} = \pi - \phi_T \tag{20}$$

The opponent's angle off is not used in this simulation.

III. Situation Cell and Tactics Selection

This chapter contains a discussion of the situation cell and maneuver selection as determined in FUNCTION ISTATE. The situation space dichotomization logic uses the four variables, range, R_F, range rate, $\dot{R}_F$, the deviation angle or steering error of the target, θ_T, and the angle off of the target, ϕ_T, which were developed in the previous chapter.

Role Designation and Maneuver Selection

An aircraft in the air-to-air combat environment may assume one of four basic roles:

1. The attack role occurs when the target is in a situation nearing weapons utilization. In this case, the deviation angle is small whether in the front or rear hemisphere of the opponent.

2. The offensive role occurs when the target has an advantage over the opponent, but is not necessarily in a position to utilize his weapons.

3. The defensive role occurs in two cases; in both cases the target is in the front hemisphere of the opponent. Either the target no longer considers himself to have a steering error advantage over the opponent, or the opponent has a steering error of less than 20 degrees.

4. The evasive role occurs when the target is in a position to have weapons launched against it. In this case, an immediate evasive maneuver must be performed.

Within each of these role designations, the target may select from several combat maneuvers, depending upon which situation cell he

occupies. Each situation cell is designated by a three digit number. The hundreds digit corresponds to the air combat role in which the target is involved.

5 - attack

4 - offense

3 - defense

2 - evasion

Within the attack role there are two basic components which correspond to attack from the front of the opponent and attack from the rear of the opponent. Likewise, the offensive role is divided into front offense and rear offense. A flow chart of the decision logic is presented in Fig. 7.

The role decision parameters used in the FUNCTION ISTATE are input from SUBROUTINE SKILL. These parameters are defined and their nominal values given below.

ϕ_{D_2} is nominally 90°. It is the target's maximum angle off for a rear attack.

α_{max} is the maximum angle of attack utilized by the target aircraft. This determines the amount of steering error which can be cancelled in an immediate attack.

θ_{D_4} is the target's maximum steering error for a frontal attack. Nominally, it is 30°.

θ_{D_5} is the target's maximum steering error for frontal offense. Nominally, it is 60°.

ϵ_{D_6} is an arbitrary angle beyond which the opponent's steering error is considered large. It is 20° in the nominal case.

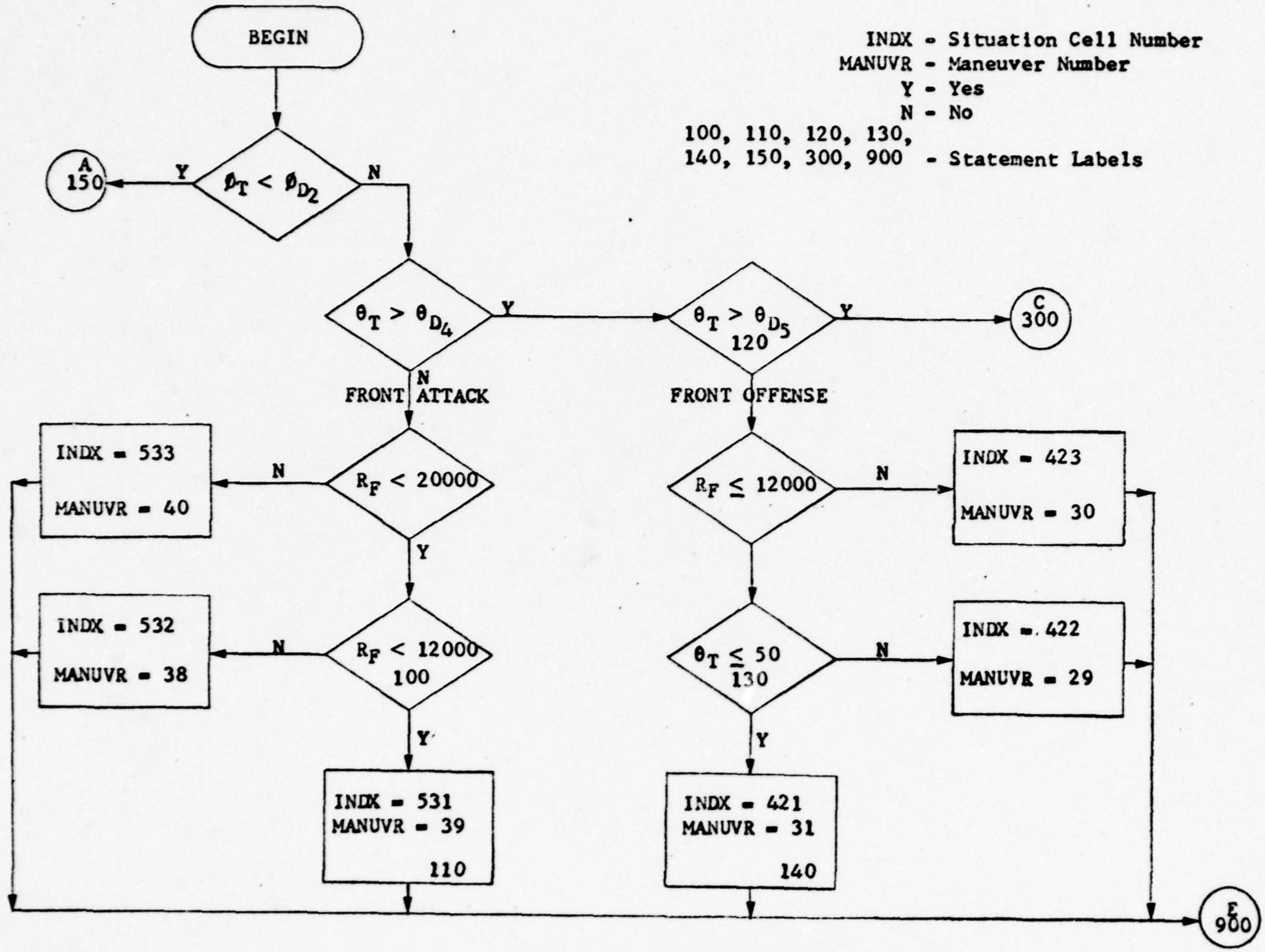


Fig. 7. ISTATE Flowchart (Ref 7:A-28)

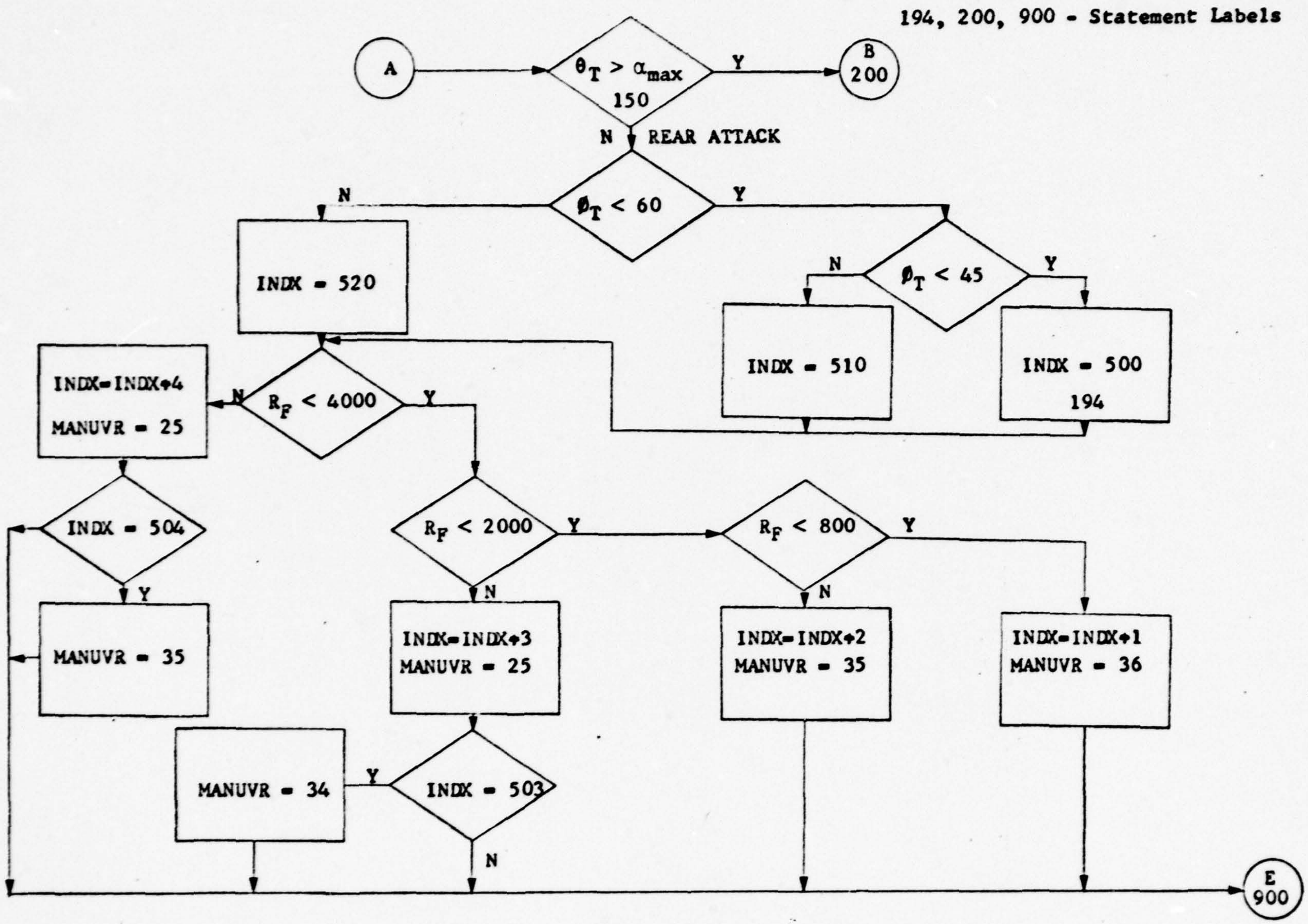


Fig. 7. (continued) ISTATE Flowchart (Ref 7:A-29)

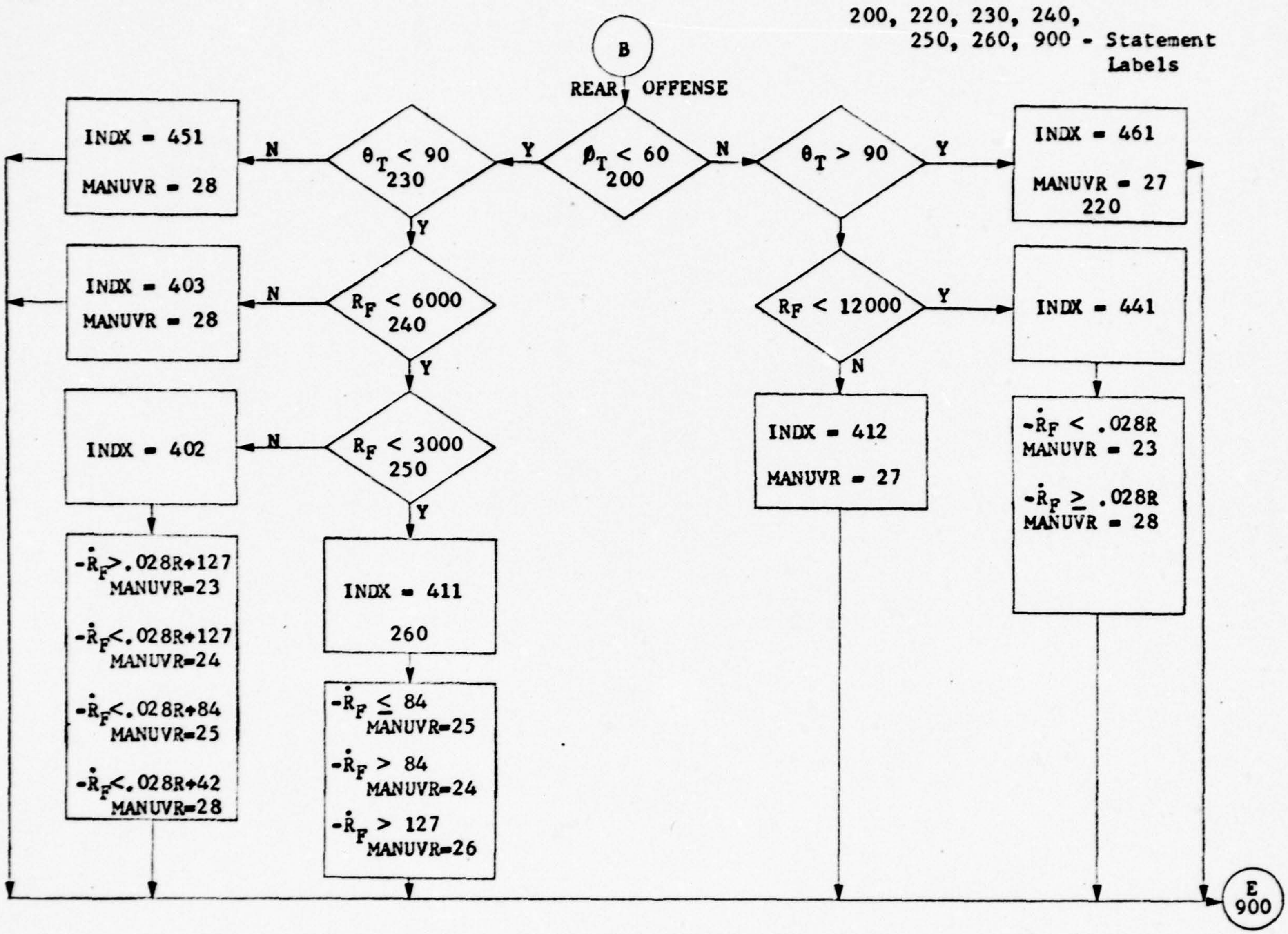


Fig. 7. (continued) ISTATE Flowchart (Ref 7:A-30)

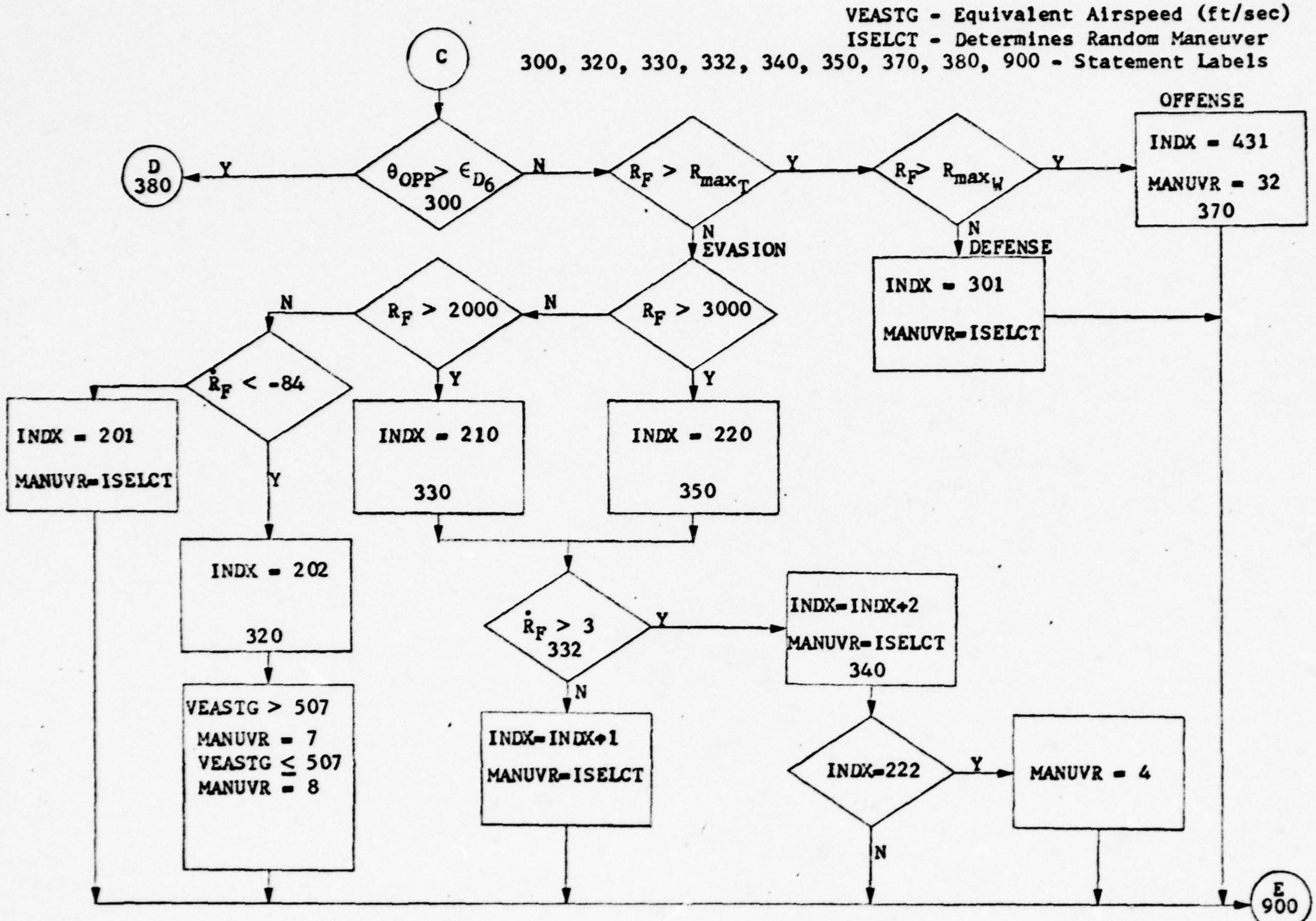


Fig. 7. (continued) ISTATE Flowchart (Ref 7:A-31)

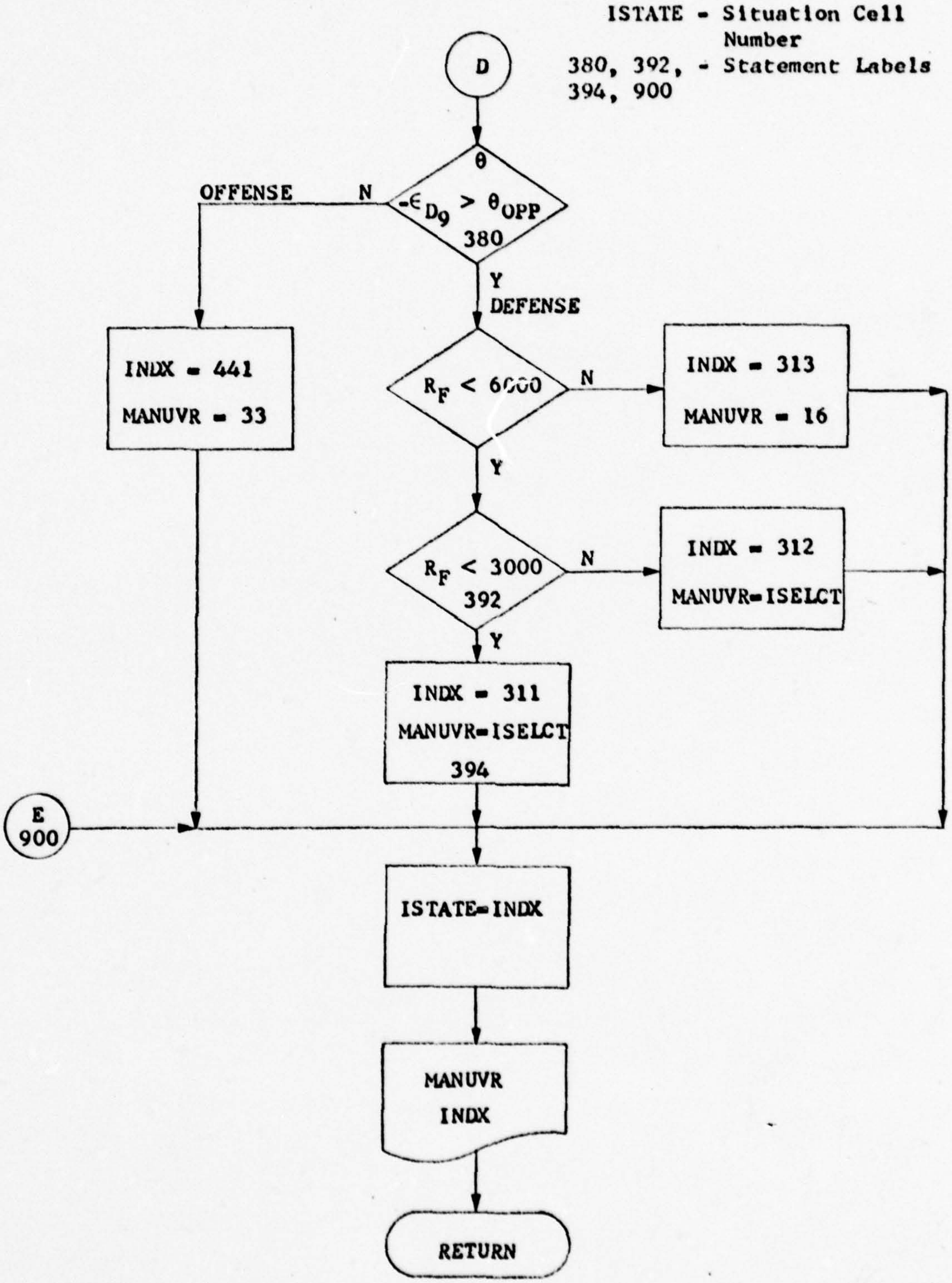


Fig. 7. (concluded) ISTATE Flowchart (Ref 7:A-32)

ϵ_{D_9} is nominally set at 10^o. If the target's steering error exceeds his opponent's steering error by more than ϵ_{D_9}, the opponent would be considered to have a distinct advantage.

R_{max_T} is the maximum effective tracking range. This indicates the range beyond which the probability of being killed by the opponent decreases sharply. If the target assumes his opponent carries missiles, R_{max_T} is arbitrarily set at 6000 feet. If the target could ascertain that the opponent had no missile capability, R_{max_T} could be set to 3000 feet. However, this would usually be an unwise assumption to make.

R_{max_W} is the maximum weapon range. Again, the target must assume the worst conditions, since he does not know the status of the weapons carried by his opponent. R_{max_W} is set at 20,000 feet. This corresponds to an opponent with a long range missile capability.

<u>Attack Role</u>

The first test in determining the target's role is whether the angle off of the target is less than ϕ_{D_2}. This indicates whether or not the target is in the front hemisphere of the opponent. The next level checks the steering error or deviation angle of the target, θ_T. If the target is in the opponent's front hemisphere and θ_T is less than θ_{D_4}, the target is in the rear attack role. On the other hand, if the target is in the front hemisphere with the steering error less than θ_{D_4}, it is in the front attack role. Once it has been determined that the target is in the front attack role the remainder of the logic is based only on range.

For the frontal attack, if the range to the opponent is greater than 20,000 feet, the target will perform a barrel roll attack. If the

range lies between 12,000 feet and 20,000 feet, the target will fly a pure pursuit for a missile attack. And if the existing range is less than 12,000 feet, a head on gun attack is appropriate.

For the rear attack role, there are three categories of ϕ_T, angle off, which are considered in addition to range. These categories of angle off are less than 45°, greater than 45° and less than 60°, or between 60° and ϕ_{D_2}. For these categories, the target will select a pure pursuit course to decrease range, a pure pursuit course to track for a missile attack, a lead pursuit course to track for a gun attack, or a high speed yo-yo to avoid overshooting the opponent. See Fig. 8 for the exact situation cells corresponding to attack role. In this figure and the ones that follow, the opponent velocity vector is located at the center of the figure and the range along the vertical axis. Although these plots are only a two dimensional representation of range and angle off, they facilitate understanding which regions apply to each situation cell.

Offensive Role

If it is determined that the target is in the rear hemisphere of the opponent, but not in the attack role, the target is in the rear offense role. In this role it is first determined whether the angle off is less than 60° or between 60° and ϕ_{D_2}. The next step is to determine whether the target has a steering error greater than 90°. This indicates that the aircraft are separating, probably as a result of a pass in a nose quarter attack. These situations are considered separately. The other rear offense situations are further broken down by range. See Fig. 9 for the rear offense situation space.

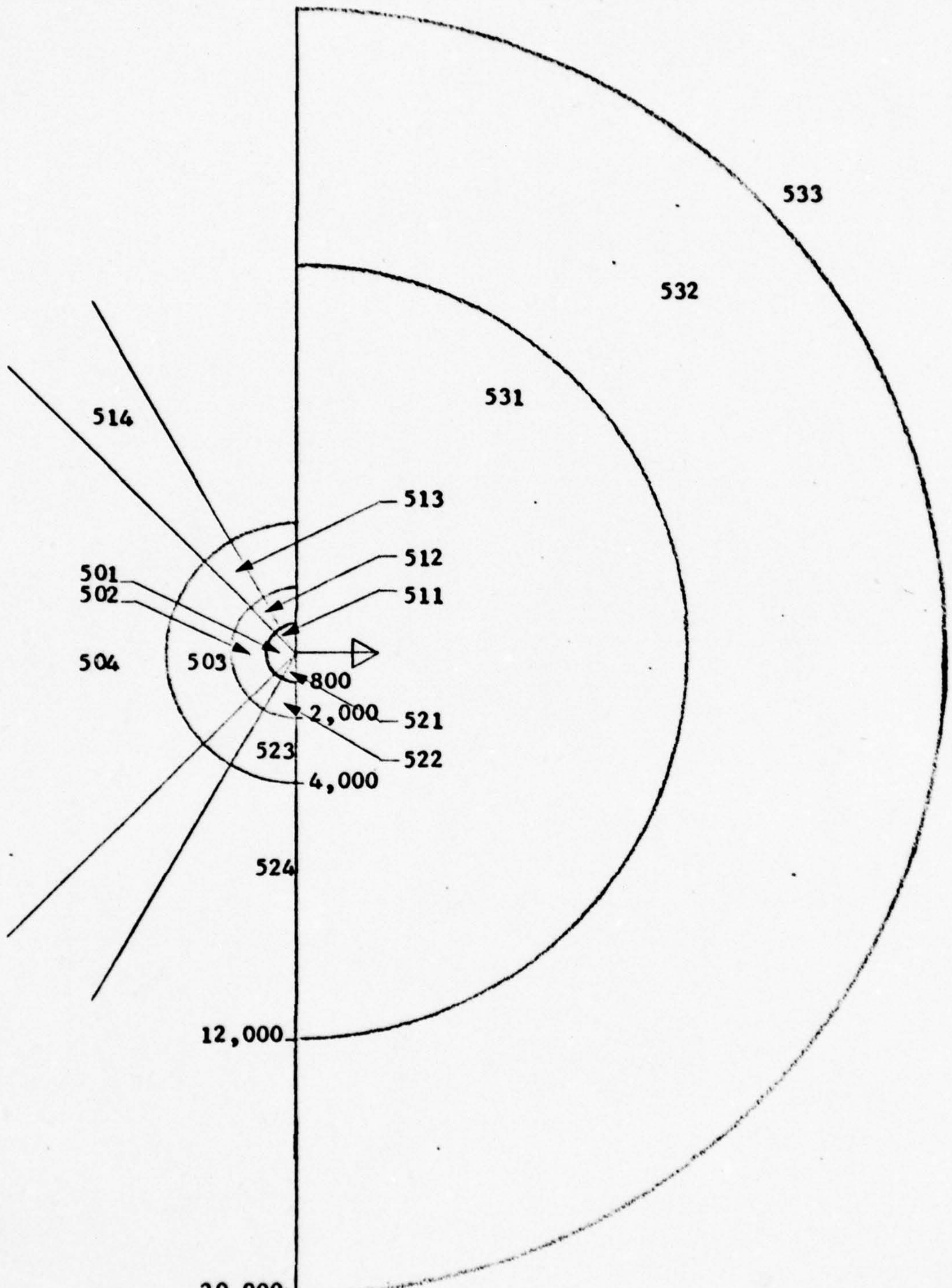


Fig. 8. Situation Space for the Attack Role (Ref 7:3-14)

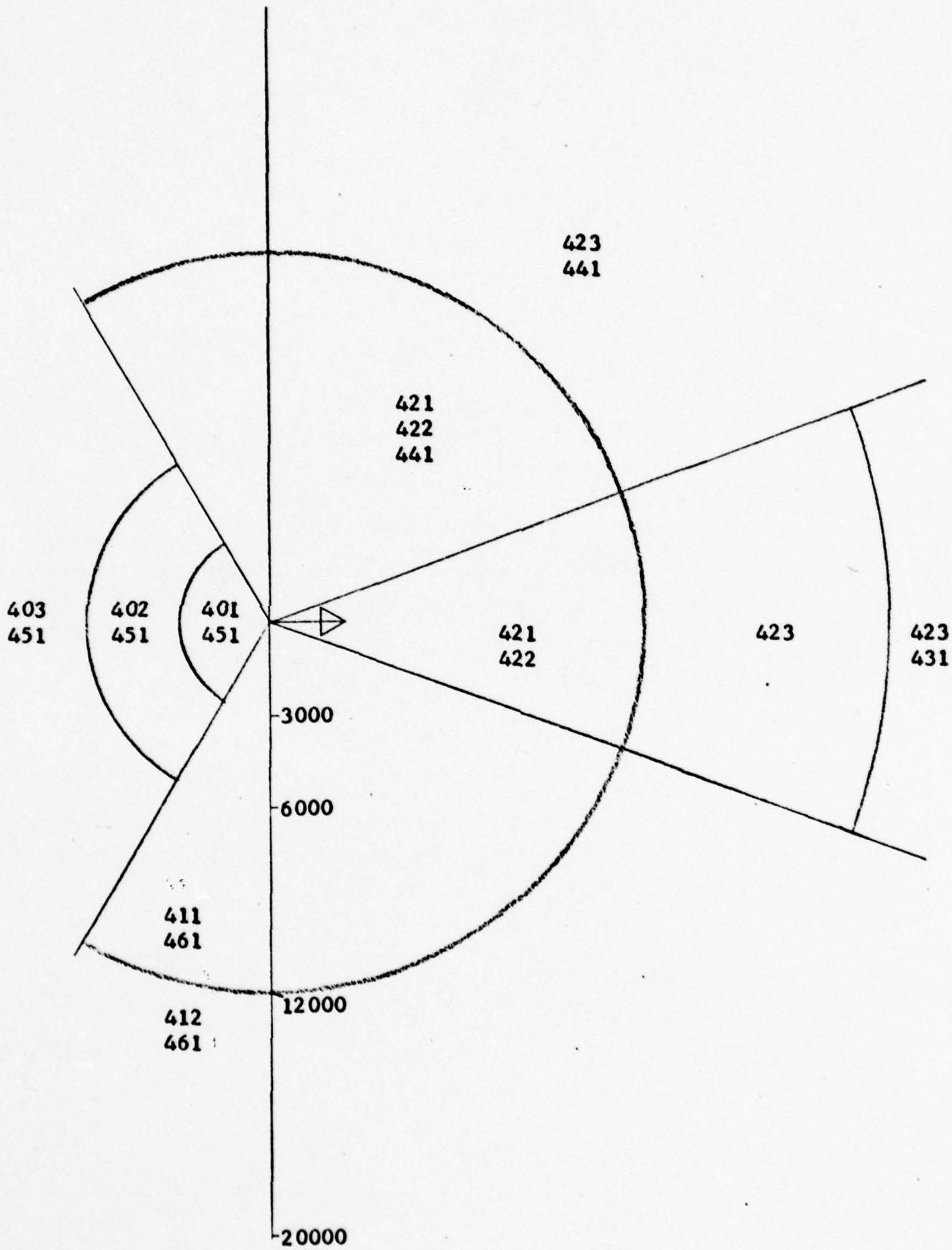


Fig. 9. Situation Space for the Offense Role (Ref 7:3-15)

There are three situations in which the target is considered to be on offense in the opponent's front hemisphere. The first case occurs when the target steering error is greater than θ_{D_4} and less than θ_{D_5}. The second case occurs when the steering error is greater than θ_{D_5} and less than ϵ_{D_9}. The third case occurs when the steering error of the opponent is less than θ_{D_4} but the range is greater than R_{max_W}. See Fig. 9.

Defensive Role

The defensive role occurs for two cases when the target is in the front hemisphere of the opponent and has a steering error greater than θ_{D_5}. The first case occurs when the target's steering error is greater than ϵ_{D_9}. The second case occurs when the range is between R_{max_T} and R_{max_W} with the opponent having a steering error less than ϵ_{D_6}.

For the first case, if the range is greater than 6000 feet, the maximum energy maneuver is selected. If the range decreases below 6000 feet, the target selects randomly between a hard turn, maximum rotation of the opponent's line-of-sight vector, or a maximum rotation of the opponent's proportional pursuit vector. These maneuvers all correspond to a hard turn into the opponent. If range decreases below 3000 feet, the target will select randomly between a turn at maximum rate, maximum rotation of the opponent's lead pursuit vector, a split s, a vertical rolling scissors, a maximum rotation of the opponent's proportional pursuit vector, or a scissors.

For the second case of the defensive role, the target will select from a maximum rate turn, a hard turn into the opponent, a maximum rotation of the opponent's line-of-sight vector, or a maximum energy maneuver to disengage. Fig. 10 shows the situation space for the defensive role.

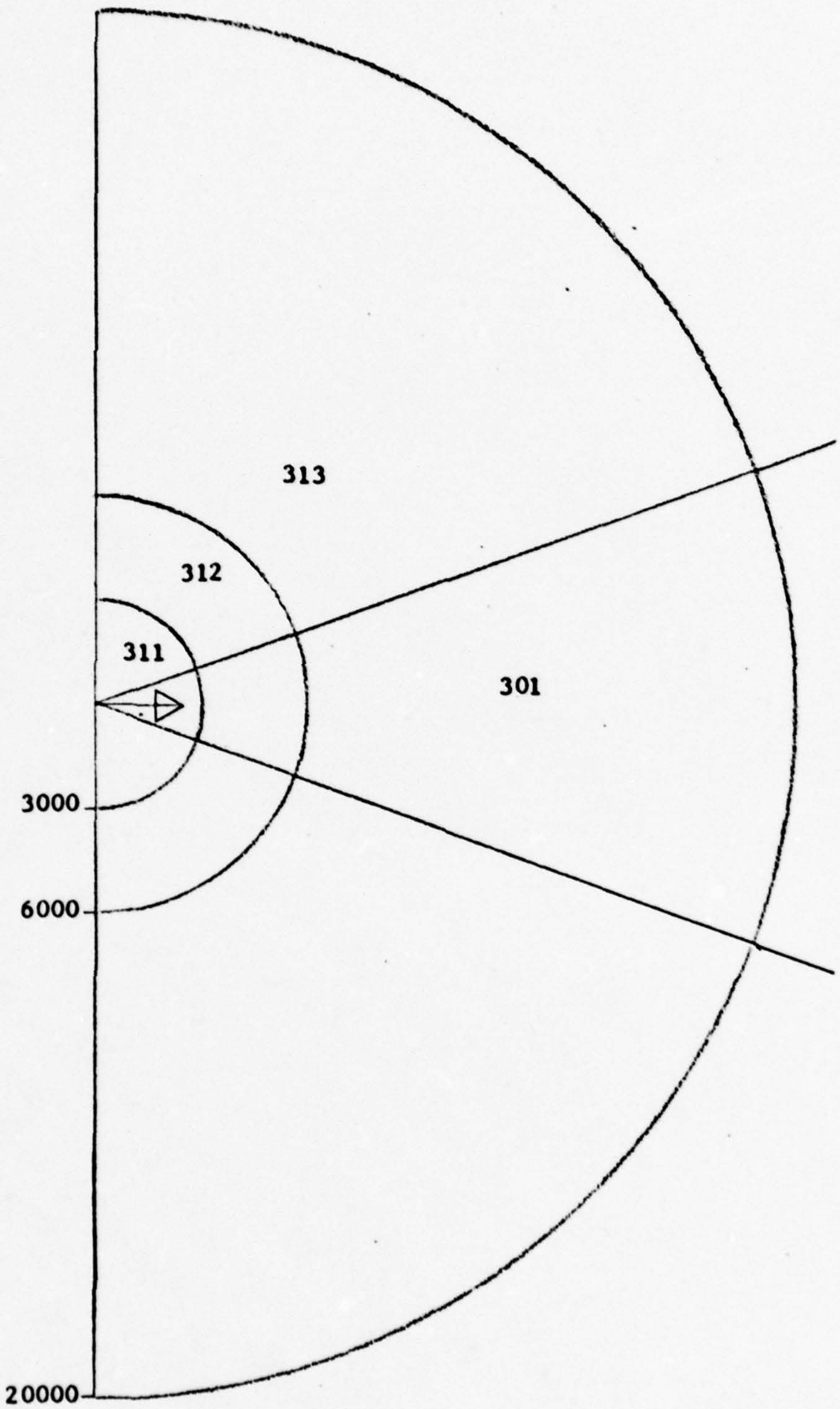


Fig. 10. Situation Space for the Defense Role (Ref 7:3-16)

Evasive Role

The evasive role occurs only when the opponent's steering error is less than ϵ_{D_6} and the target is less than the range, R_{max_T} from the opponent.

Within the evasive role, the opponent is inside the maximum tracking range for his weapon. If, however, the range is greater than 3000 feet and is increasing, the target should capitalize on this energy advantage to increase the separation distance. This is accomplished by performing a maximum energy maneuver.

If the target does not possess an energy advantage, the target has several maneuvers which are appropriate. These are a hard turn into the opponent, a vertical dive followed by a hard pull-up, or a maximum rate turn. Each of these maneuvers is appropriate, and one is selected at random, with a better chance of selecting a hard turn into the opponent.

As the opponent comes inside 3000 feet range, the split S is added to those above in the target's list of choices. When the opponent comes inside 2000 feet, he is in gun firing position and the target must immediately increase the opponent's angle off. Two maneuvers are added. They are an immediate pull-up at maximum angle of attack and a hard turn followed by a turn reversal. The target will no longer consider the maximum rate turn or the vertical dive followed by a hard pull up. In both of these maneuvers, the target must decrease his angle of attack, and does nothing to immediately increase the opponent's angle off.

With the range less than 2000 feet, and an excessive rate of closure, the target should perform either a high g roll over or a high g roll underneath. The decision is based upon his own airspeed.

If his airspeed is below 350 KCAS, where KCAS is knots calibrated airspeed, he will never be able to roll over the top, and would perform the roll underneath. See Fig. 11 for the evasive role situation space.

Associated with each situation cell are one or more appropriate maneuvers. When the target occupies a defensive or evasive cell, the choice of maneuvers is made on a random basis in such a manner that a particular maneuver is selected a predetermined fraction of the time. Certain offensive cells also have more than one maneuver. However, these maneuvers are chosen on a deterministic basis. Table I contains a list of the maneuvers associated with each situation cell.

The final output of FUNCTION ISTATE consists of the situation cell number and the selected maneuver for that situation cell. These two parameters are then utilized by SUBROUTINE DESIRE to generate the actual controls needed to simulate the maneuver. See Appendix A for a detailed listing of FUNCTION ISTATE.

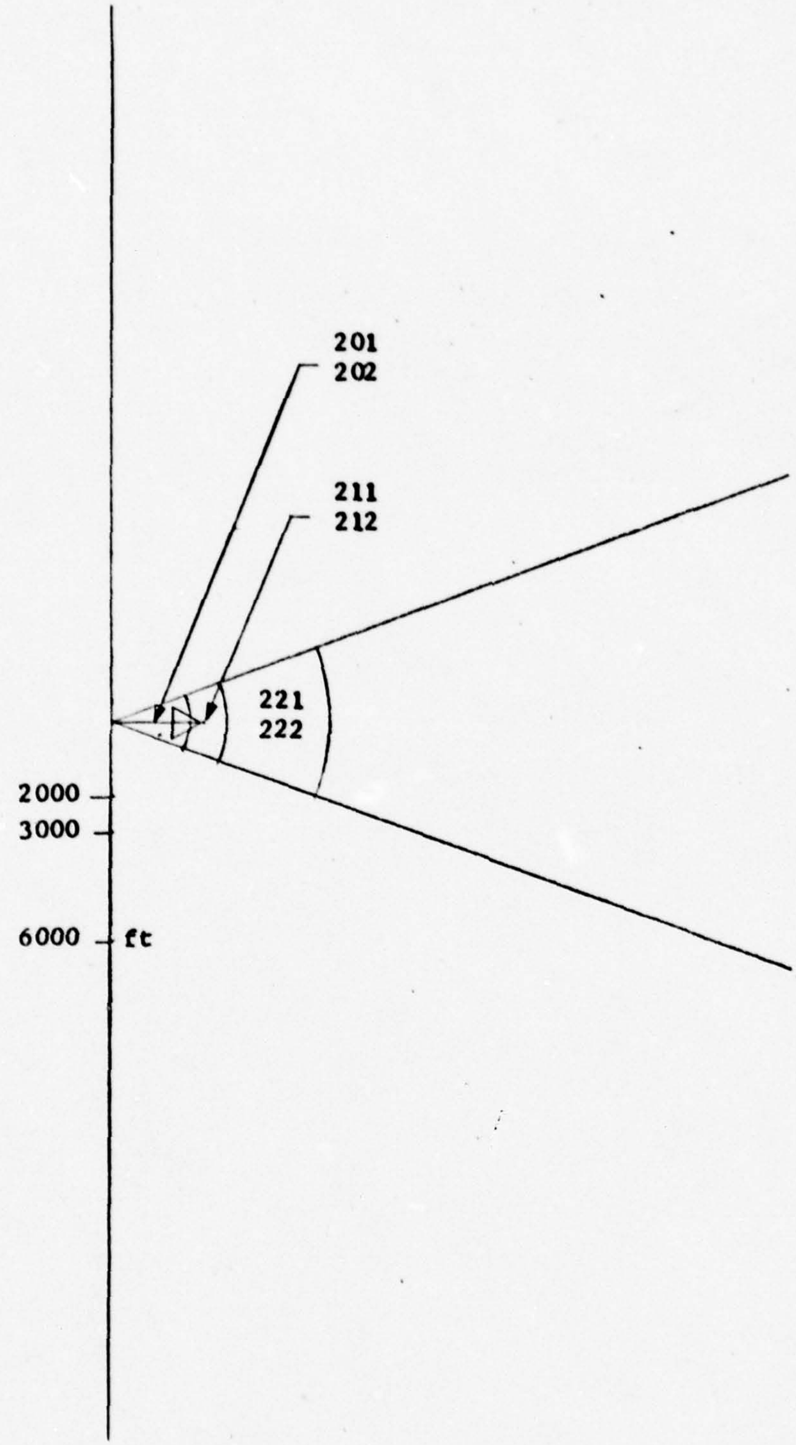


Fig. 11. Situation Space for the Evasive Role (Ref 7:3-17)

Table I
Air Combat Maneuvers Selected for Each Situation Cell

Situation Cell	Maneuver Selected	Random Selection Rate/ Selection Criteria
201	1. Hard pull up	30%
	2. Split s	20%
	3. Hard turn	30%
	4. Hard turn followed by a reversal	20%
202	1. High g roll over	Equivalent airspeed > 507 feet/sec
	2. High g roll underneath	Equivalent airspeed $\leq$ 507 feet/sec
211	1. Hard turn	60%
	2. Maximum rate turn	20%
	3. Vertical dive followed by hard pull up	20%
212	1. Split s	20%
	2. Hard turn	40%
	3. Maximum rate turn	20%
	4. Vertical dive followed by hard pull up	20%
221	1. Hard turn	60%
	2. Maximum rate turn	20%
	3. Vertical dive followed by hard pull up	20%
222	Maximum energy maneuver	
301	1. Maximum rate turn	40%
	2. Hard turn	40%
	3. Maximum energy maneuver	20%
311	1. Hard turn	60%
	2. Split s	20%
	3. Scissors	10%
	4. Vertical rolling scissors	10%
312	Hard turn	
313	Maximum energy maneuver	
401	1. High speed yo-yo	$\dot{R} < -127$
	2. Pure pursuit	$\dot{R} < -84$
	3. Lead pursuit	$\dot{R} \geq -84$

Table I (continued)
Air Combat Maneuvers Selected for Each Situation Cell

Situation Cell	Maneuver Selected	Random Selection Rate/ Selection Criteria
402	1. Low speed yo-yo	$-\dot{R} < .028R + 42$
	2. Lead pursuit	$-\dot{R} < .028R + 84$
	3. Pure pursuit	$-\dot{R} < .028R + 127$
	4. Lag pursuit	$-\dot{R} > .028R + 127$
403	Low speed yo-yo	
411	1. Lag pursuit	$\dot{R} > -.028R$
	2. Low speed yo-yo	$\dot{R} < -.028R$
412	Barrel roll	
421	Hard turn	
422	Pure pursuit	
423	Barrel roll	
431	Barrel roll	
441	Pure pursuit	
451	Low speed yo-yo	
461	Barrel roll	
501	High speed yo-yo	
502	Lead pursuit (gun attack)	
503	Pure pursuit (missile attack)	
504	Pure pursuit	
511	High speed yo-yo	
512	Lead pursuit (gun attack)	
513	Lead pursuit	
514	Lead pursuit	
521	High speed yo-yo	
522	Lead pursuit (gun attack)	
523	Lead pursuit	

Table I (continued)
Air Combat Maneuvers Selected for Each Situation Cell

Situation Cell	Maneuver Selected	Random Selection Rate/ Selection Criteria
524	Lead pursuit	
531	Head on gun attack	
532	Pure pursuit	
533	Barrel roll	

IV. Air Combat Maneuvers

The control inputs used to simulate each Air Combat Maneuver are contained in SUBROUTINE DESIRE. The main program calls DESIRE on each time through the loop. As a result, the desired control inputs are continuously updated.

The controls utilized in this simulation are: α_D, angle of attack, β_D, sideslip angle, ϕ_D, bank angle, and $Thrust_D$, thrust. These correspond to the pilot's control over elevator, rudder, ailerons, and throttle, but eliminate the need of simulating a complicated control system.

In spite of the development of high performance fighter aircraft, pilots today learn the same basic air combat maneuvers that have been flown since World War I. The only changes have been in the size of the arena and in the formation tactics utilized. Since the success of Smart Target depends a great deal upon the maneuvers being realistic, it is appropriate to discuss the purpose of each maneuver and its desired outcome in air battle.

Evasive Maneuvers

Maximum Energy Maneuver. The maximum energy maneuver may be performed in any phase of air combat, but is normally considered as an evasive or defensive maneuver. As its name implies, this maneuver is designed to increase the aircraft's velocity and thus increase its energy level. The purpose of this maneuver is to increase the separation distance between aircraft or to exit from the combat arena. By simply unloading the aircraft at 0 - .5 g's, in a wings level attitude, with

maximum thrust, the pilot is able to increase his energy level and accelerate to a higher mach number than his opponent.

Hard Turn. The purpose of the hard turn is to prevent the opponent from achieving a firing position. The object of this maneuver is to rotate the target's angular velocity cone away from the opponent. The main concern when performing this hard turn is to acquire a smaller turn radius than the opponent. This will force him outside the target's turn and prevent the opponent from achieving a tracking solution. The hard turn in this simulation is performed in the horizontal plane.

Vertical Dive Followed by a Hard Pull Up. This maneuver accomplishes the same objectives as the hard turn, except that it is flown in the vertical rather than the horizontal plane. The target rolls to an inverted attitude and pulls the aircraft into a dive. As the dive nears the vertical position the target will roll wings level and pull up with maximum angle of attack. If the opponent has followed the target into the dive, the hard pull up is designed to force the opponent to overshoot the bottom of the target's flight path and thus break any tracking solution which the opponent possessed. See Fig. 12 for a depiction of the vertical dive followed by a hard pull up.

Hard Pull Up. The hard pull up serves the same purpose as the vertical dive followed by a hard pull up, but is done at closer range when there is insufficient time to accomplish the dive. It also attempts to force the opponent to overshoot the target's flight path and preclude a tracking solution.

Maximum Rate Turn. This maneuver is comparable to the hard turn, but is accomplished at greater range. The maximum roll rate allows the pilot to very quickly obtain his desired bank angle. As the turn is

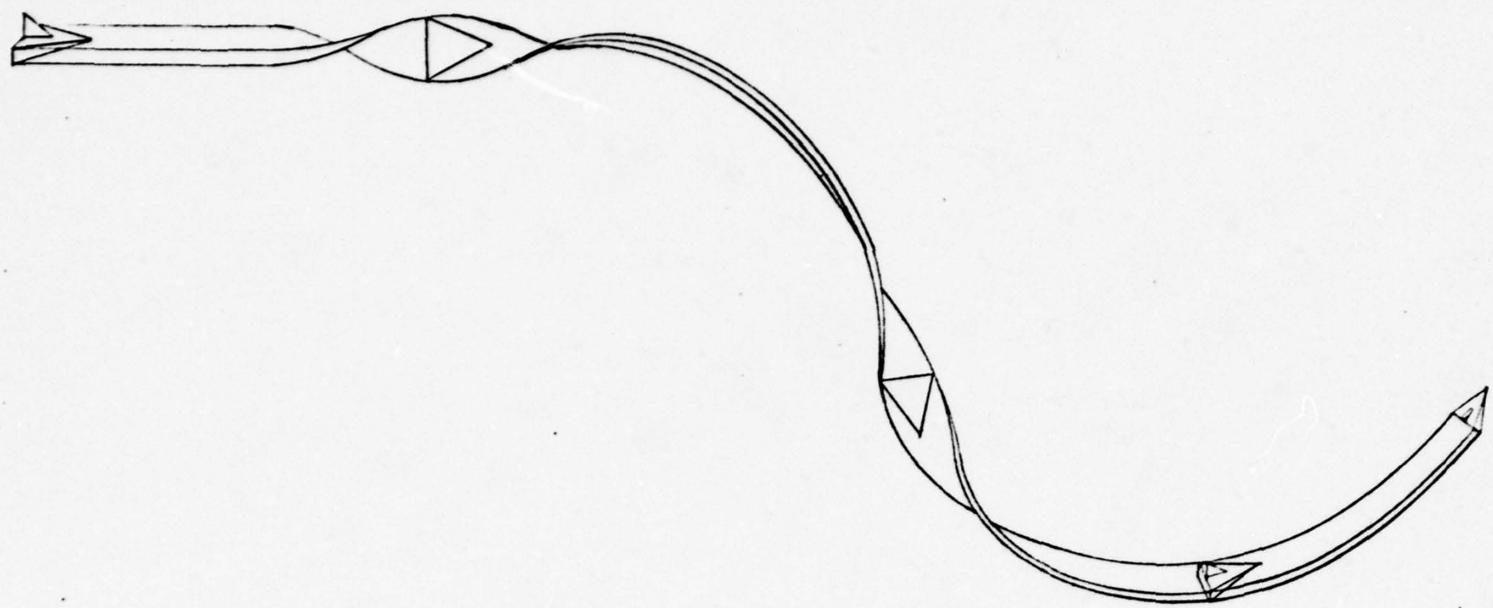


Fig. 12. Vertical Dive Followed by a Hard Pull Up

started, a slight dive is employed to generate angular velocity and also to retain future maneuvering potential. Because the opponent is at a greater range, his rate of turn will appear to be less than the target's, and his angle off and rate of closure will increase. Again, the desired result is to cause the attacking aircraft to overshoot the target's flight path.

Hard Turn Followed by a Turn Reversal. The hard turn with reversal, like the other evasive maneuvers, is designed to defeat an opponent's tracking solution. In this maneuver, the target will randomly fly a hard turn, followed by a turn in the opposite direction. The hard turns make it extremely difficult for the opponent to achieve consistent tracking due to the unpredictability of the maneuver.

Split S. The Split S is initiated identically to the vertical dive followed by a hard pull up. The maneuver continues to be the same until the vertical position is achieved. Instead of rolling at this point, the target will continue to increase back pressure, flying the aircraft through the vertical and using the pull out from the dive to cause the opponent to overshoot. Figure 13 shows the flight path of a Split S.

High G Rolls. The high g rolls are very effective evasive maneuvers at close range because they are difficult to counter effectively. These rolls appear to be simple aileron rolls, but in reality are high performance vector rolls performed at high angles of attack, which allow the defender to utilize gravity and induced drag to reduce his airspeed and change direction more rapidly than his opponent. The objective of these maneuvers is to remove the attacker from his firing position. Figs. 14 and 15 depict the high g roll over and the high g roll under, respectively.

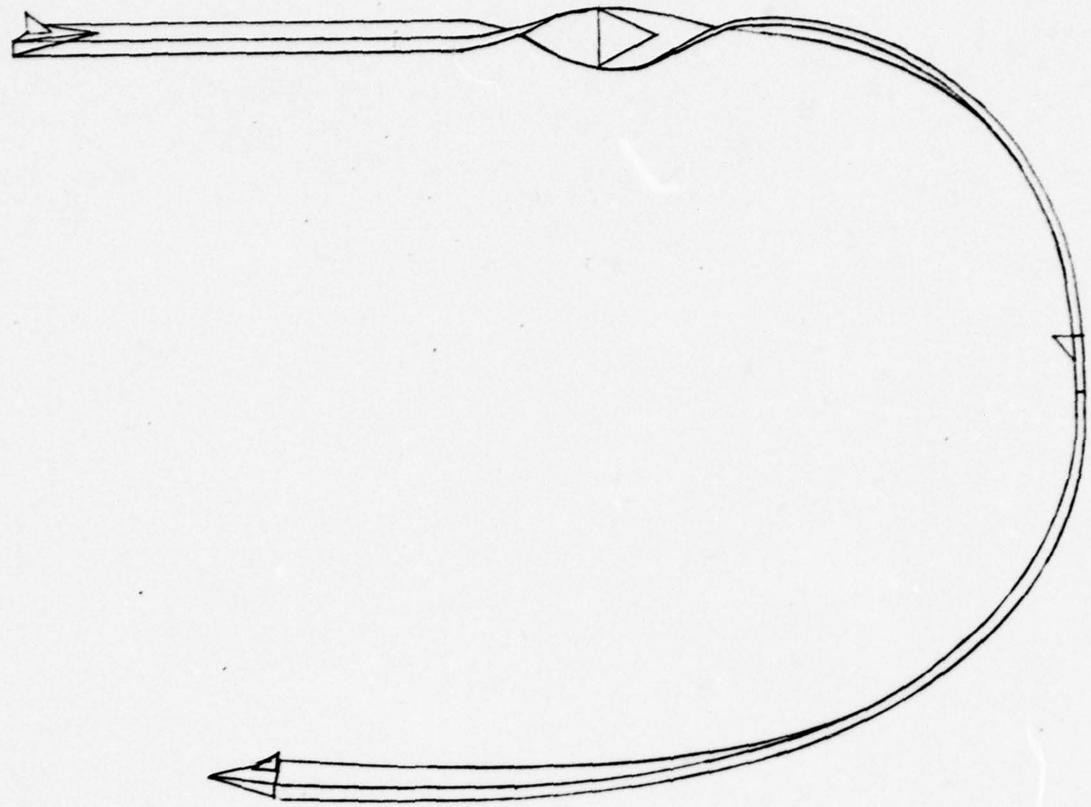


Fig. 13. Split S Maneuver

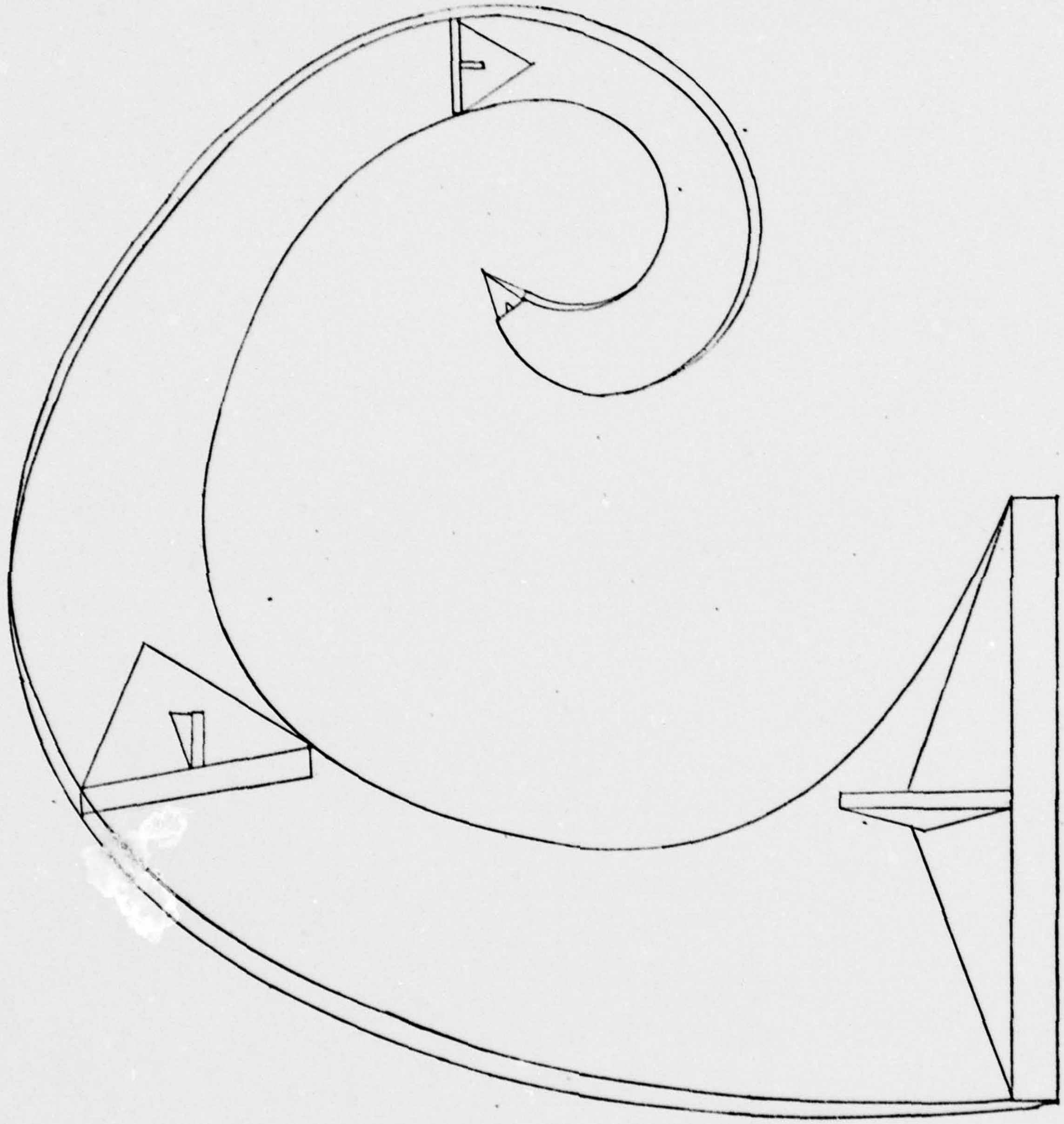


Fig. 14. High G Roll Over

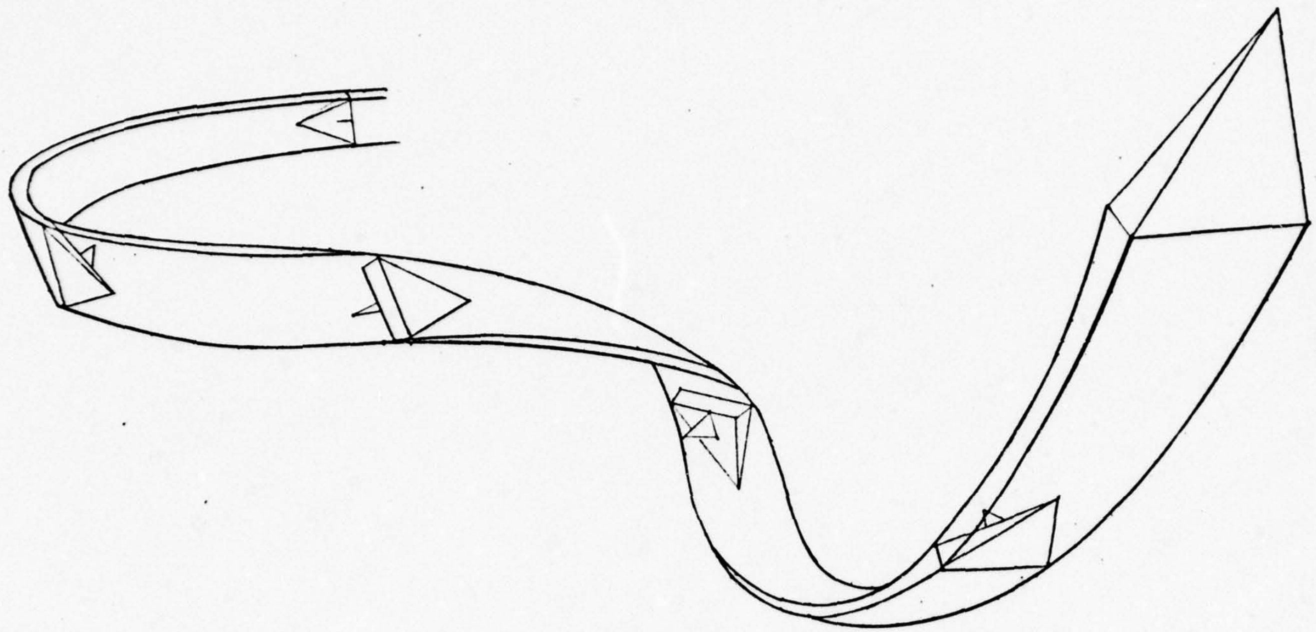


Fig. 15. High G Roll Underneath

Defensive Maneuvers

Many of the defensive maneuvers are the same as those employed in the evasive role. These are the maximum energy maneuver, hard turn, split S and maximum rate turn. In addition to these maneuvers, the scissors, and vertical rolling scissors are employed as defensive maneuvers.

Scissors. The scissors is a defensive maneuver in which a series of turn reversals is executed in an attempt to achieve an offensive position after an overshoot by the attacker. In this maneuver, the target has the advantage. By virtue of forcing the attacker to overshoot, the target has a lower velocity and can easily force the attacker to the target's twelve o'clock position. See Fig. 16 for a depiction of the scissors maneuver. In Fig. 16, and the remaining figures in this chapter, the relative positions of the two aircraft, in time, are denoted by identical alphabetic letters.

Vertical Rolling Scissors. This is a defensive rolling maneuver in the vertical plane. The purpose of this maneuver is to gain an offensive advantage if the attacker overshoots the target's flight path, and slides through his angular velocity cone while in the vertical plane. See Fig. 17.

Offensive Role

Pursuit Curve. A pursuit curve is an offensive maneuver along the projected flight path from the target's nose to the opponent's tail. As the offender, it is desirable to position the aircraft in a good firing position for weapons utilization. Depending on relative position and which weapon is desired, lag pursuit, pure pursuit, or lead pursuit

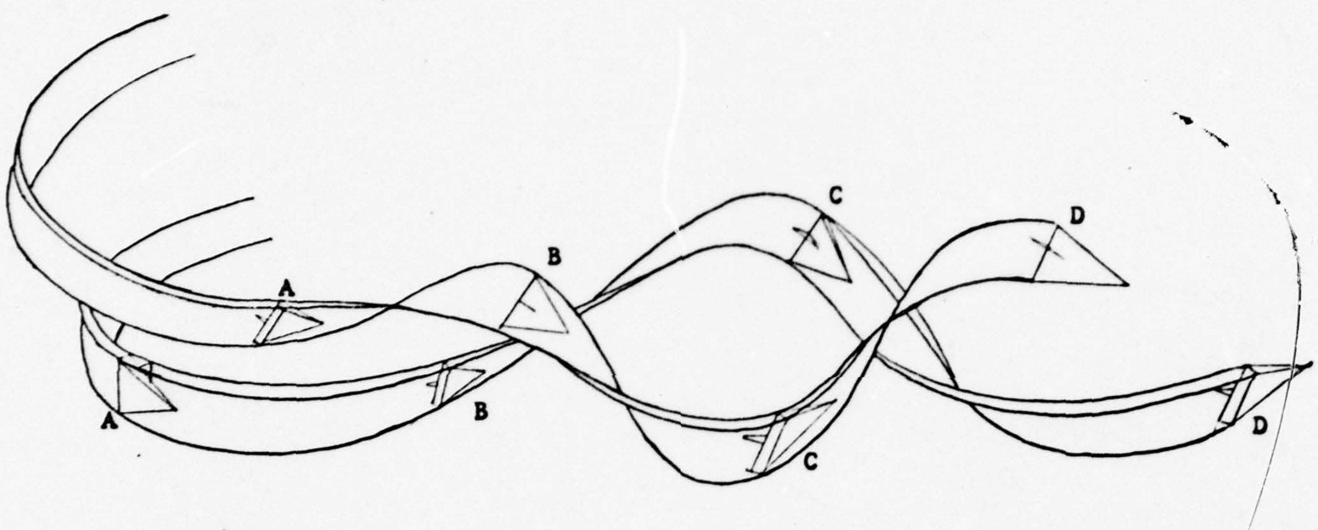


Fig. 16. Scissors Maneuver

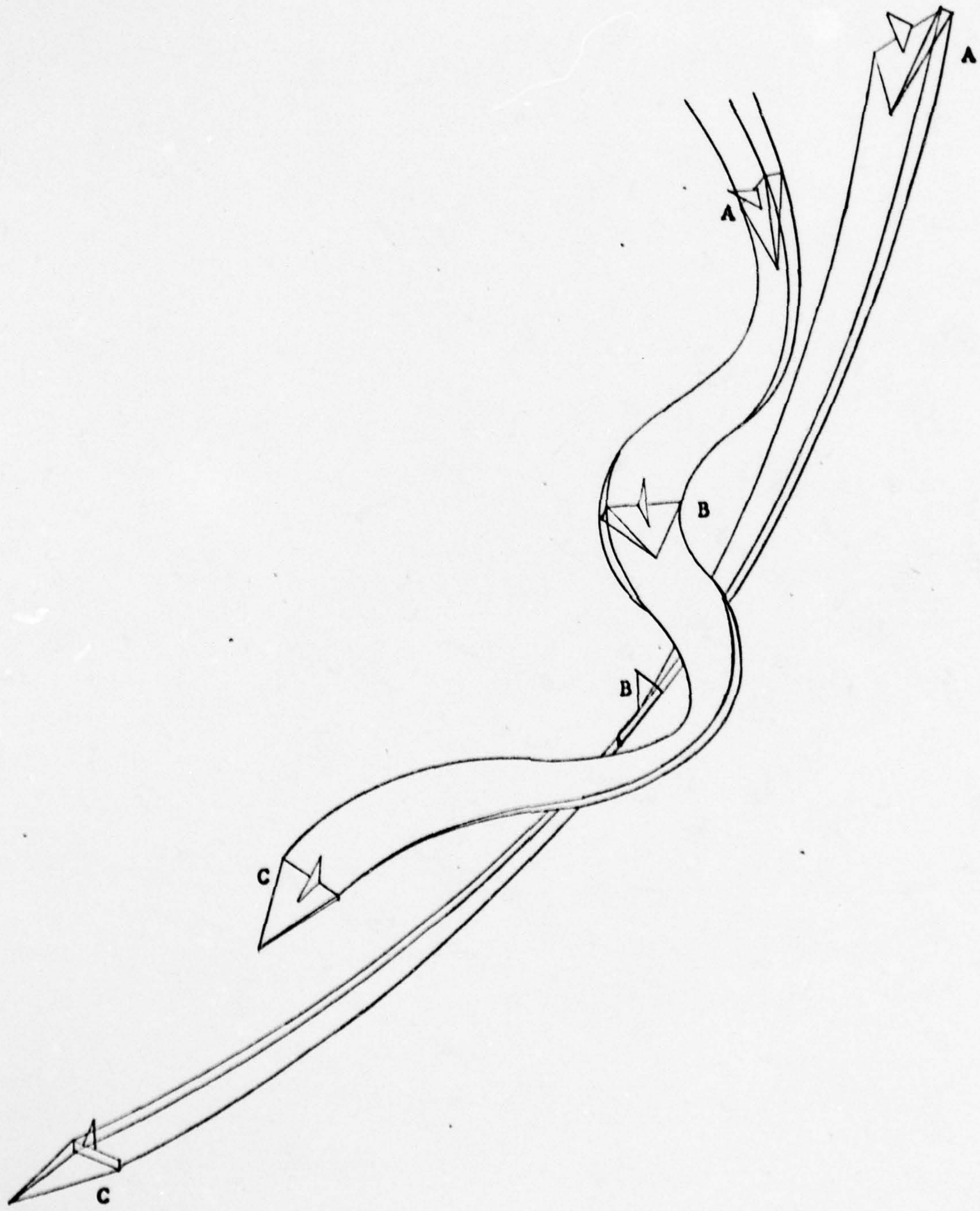


Fig. 17. Vertical Rolling Scissors

is chosen. For pure pursuit, the target is pointed directly at the opponent. For lag pursuit, the target points his nose behind the opponent. In this simulation the lag angle is 20°. Likewise, for lead pursuit, the target points his nose 20° in front of the opponent. See Fig. 18 for the pure pursuit curve.

High Speed Yo-Yo. The high speed yo-yo is an offensive maneuver in which the target maneuvers through both the vertical and horizontal planes to prevent an overshoot in the plane of the opponent's turn. The purpose is to maintain an offensive advantage by keeping nose-tail separation between the attacker and defender. The high speed yo-yo is an effective counter to the defensive turn, the scissors, and high 'g' rolls. The high speed yo-yo countering the defensive turn is shown in Fig. 19.

Barrel Roll Attack. The barrel roll attack is a three dimensional maneuver designed to decrease angle off and maintain nose-tail separation. It accomplishes the same result as the high speed yo-yo, but is normally flown at high angle off and long range. The attacker pulls up to the inside of his opponent, then barrel rolls in a direction opposite his opponent's turn. The attacker changes his rate of roll to remain inside his opponent's turn while reducing angle off and diving below his opponent's flight path. The barrel roll attack is shown in Fig. 20.

Low Speed Yo-Yo. This maneuver may be employed in a running battle or in a turning fight whenever the attacker has an insufficient rate of closure. The purpose of a low speed yo-yo is to provide cut-off and rate of closure. To perform this maneuver in a turning fight, the attacker maintains his bank and lowers his nose to the inside of the turn, thus increasing airspeed and reducing angle off. As the attacker

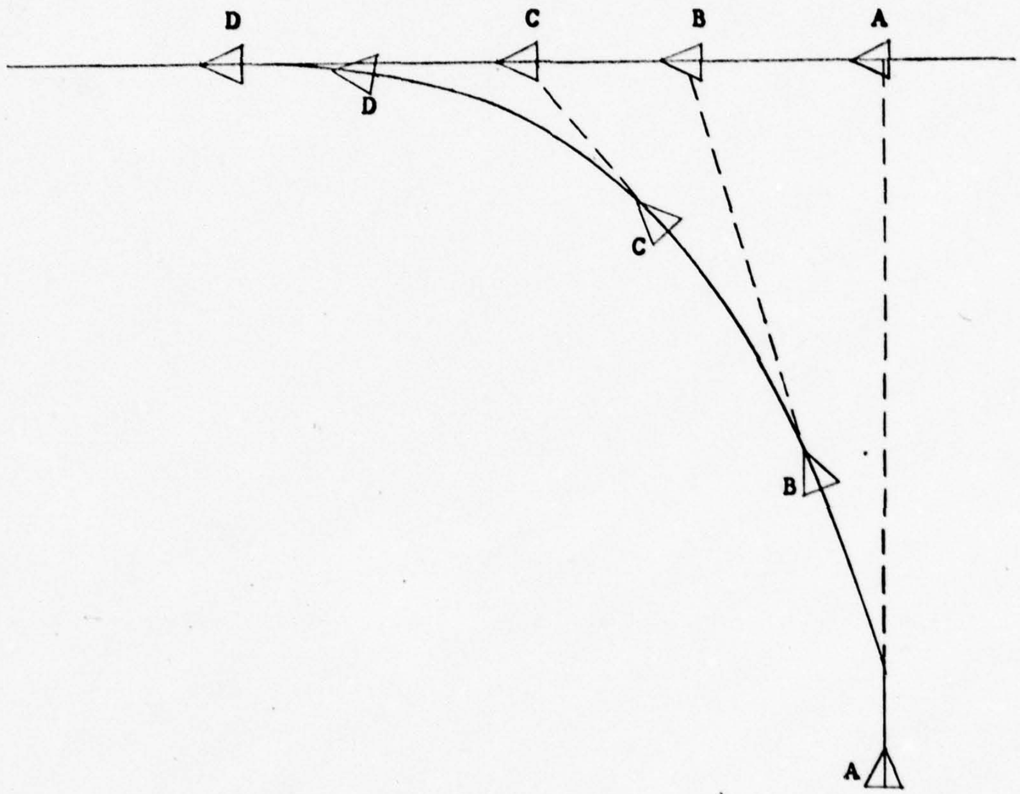


Fig. 18. Pure Pursuit Curve

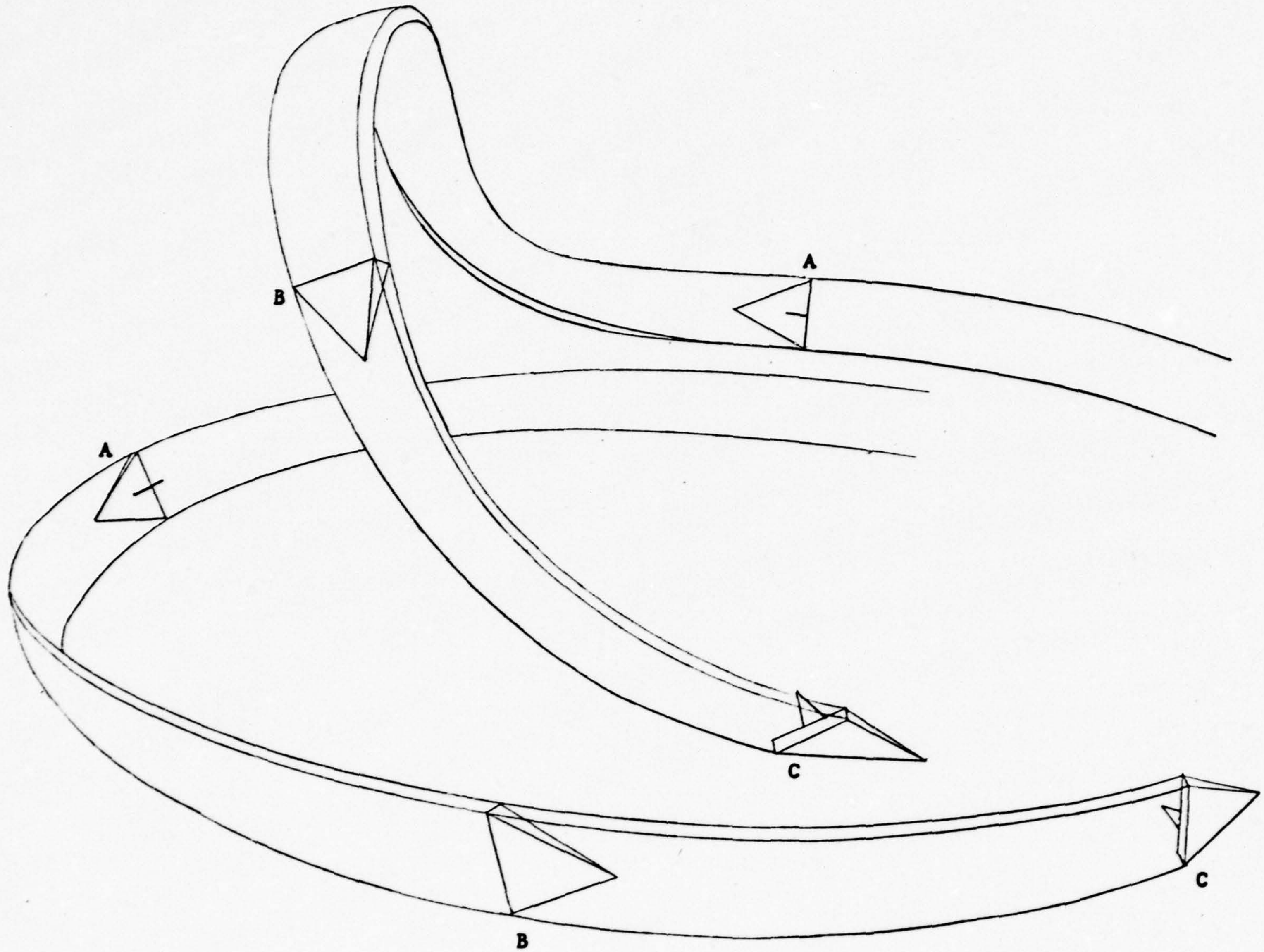


Fig. 19. High Speed Yo-Yo

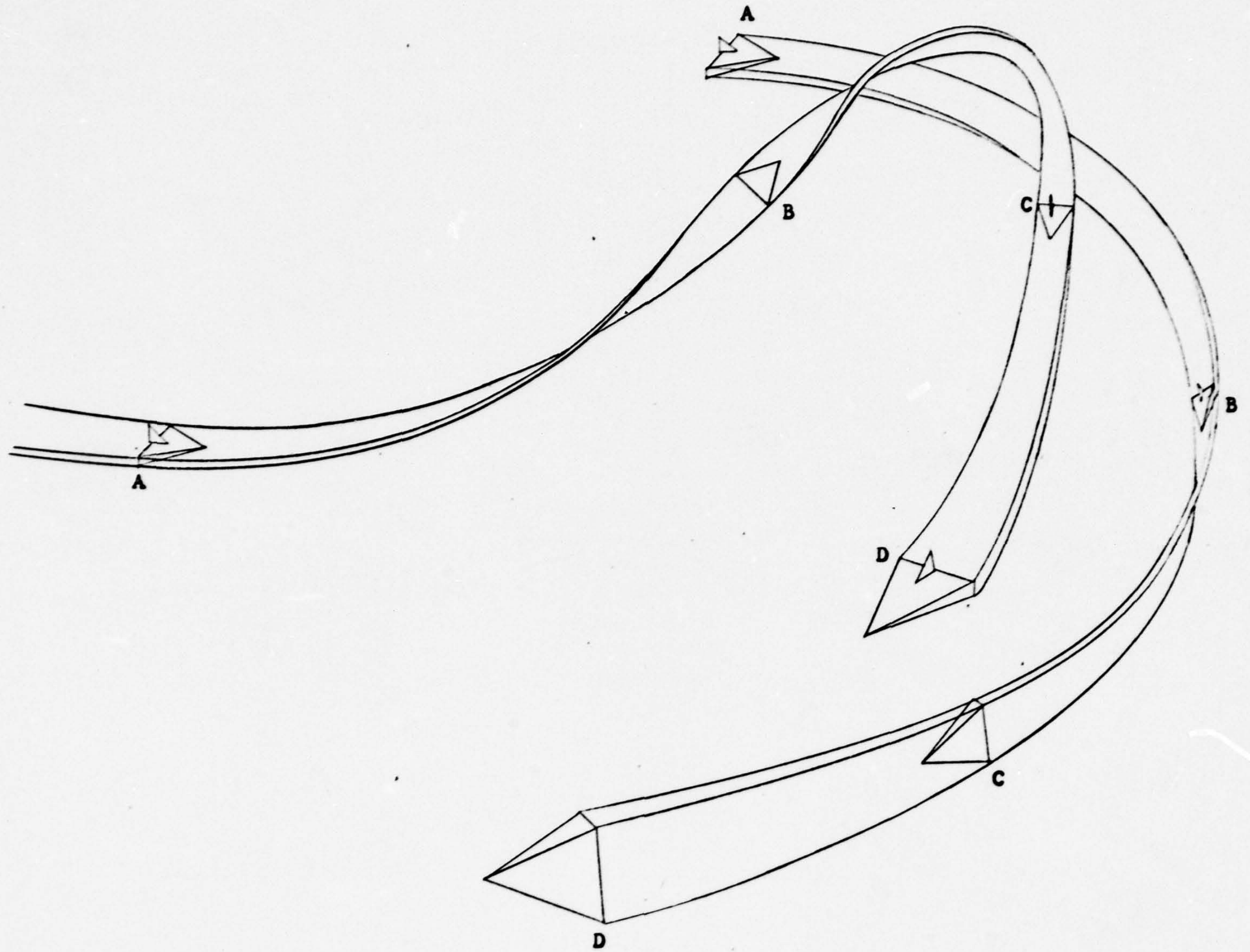


Fig. 20. Barrel Roll Attack

decreases range, with an airspeed advantage, he pulls up and zooms toward the opponent's six o'clock position. See Fig. 21.

Attack Maneuvers

Missile Attack. In performing the missile attack, the attacker simply attempts to remain on a pure pursuit course. If the attacker is able to keep his angle off less than 10° for three seconds with his relative range within the missile firing envelope, he is considered to have tracked his opponent for sufficient time to achieve a missile kill.

Gun Attack. The gun attack is accomplished by maintaining a lead pursuit angle of 20°. If the attacker maintains this lead angle, and is within 5° azimuth error and 20° elevation error, and is within gun tracking range, he has tracked the opponent for sufficient time to achieve a gun kill. Because air-to-air gunnery is based on attitude control rather than the velocity vector, these limits may be overly restrictive in achieving an actual kill.

Head-On Gun Attack. Although the head-on gun attack is not the most effective method of achieving a kill, it is a valuable psychological maneuver which may cause the opponent to commit a crucial error or force him to disengage from the combat arena. This head-on attack is accomplished by utilizing a 10° lead angle. Although the head-on attack may not achieve a kill, it will not adversely affect the target's future maneuvering potential.

It is difficult to visualize many of the maneuvers utilized in this simulation, especially if the reader has never been exposed to them or seen them flown. A more detailed description of each maneuver may be found in Ref 9 or 10. It should be emphasized that skill in air combat

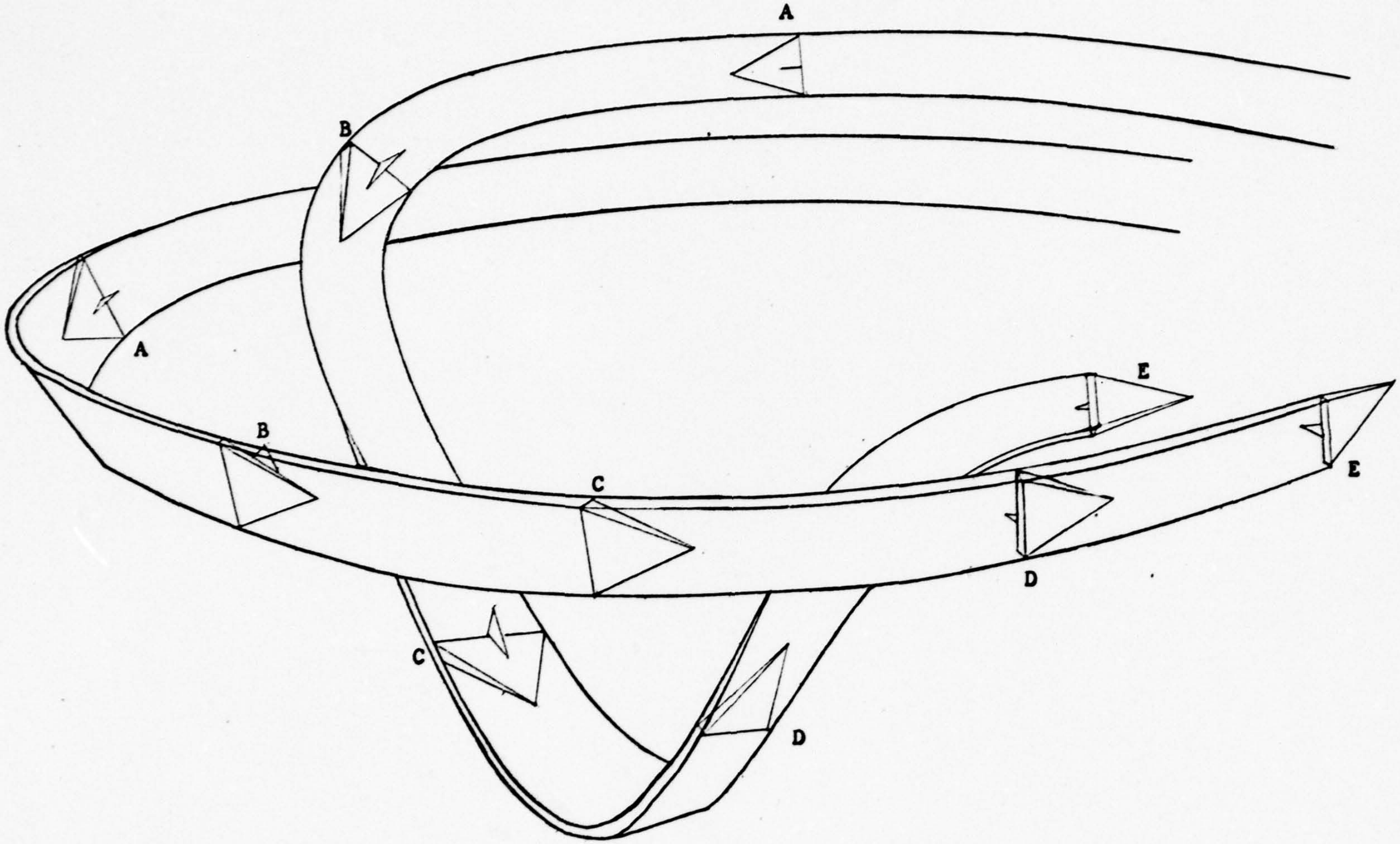


Fig. 21. Low Speed Yo-Yo

is more of an art than a science, and many decisions reached by pilots may be more accurately described as reflex reactions rather than classical decisions. Consequently, there may be those that disagree with some of the maneuvers selected in this simulation. However, it has been necessary in this project to define exact boundaries where the selection of maneuvers may change. As most fighter pilots agree, this is not the case in air-to-air combat, where even the same pilot may view identical conditions differently from day to day. It is important that the reader be aware that in reality, the maneuvers and the situations in which they are selected may differ from the situations being simulated in this study.

The desired control inputs which simulate the air combat maneuvers in this thesis are found in Appendix A, SUBROUTINE DESIRE.

V. Pilot Model

It is the purpose of the Smart Target Simulation to provide a target model which will duplicate as closely as possible the maneuvers flown by air combat pilots. It would be difficult to motivate a simulator pilot if he knew that he would always lose to a target whose performance is optimal or always win over a target whose capabilities are exceeded.

A target is desired that is less than perfect in its decision making and execution, and yet is realistic and skillful enough to make the pilot work for his victories. Because each target pilot is different, this target should have the capability to vary its skill, so that its decisions and maneuvers do not become predictable.

SUBROUTINE SKILL was developed as a very simple pilot model which allows various pilot decision boundaries to be altered. There are eight decision parameters which are functions of the pilot skill factor.

ϕ_{D_2} is the maximum angle off for rear attack.

α_{max} is the maximum angle of attack utilized by the target pilot.

θ_{D_4} is the maximum steering error for frontal attack.

θ_{D_5} is the maximum steering error for frontal offense.

ϵ_{D_6} is the steering angle error limit to determine if the opponent's steering error is small.

ϵ_{D_9} is the steering error advantage allowed the opponent.

R_{max_T} is the estimated maximum range at which the opponent may begin tracking and expect a reasonable probability of kill.

R_{max_W} is the estimated maximum effective range that the opponent may fire his weapon and possibly achieve a kill.

The nominal values of these decision parameters are listed in Table II under Pilot Skill Factor 0. These nominal values represent the decision parameters which the ideal target pilot would use.

The Real Pilot

SUBROUTINE SKILL considers a continuum of pilot skill levels. On one end of the spectrum is the pilot who has reached an experience plateau where he fails to recognize his own weaknesses. He is an over-confident, over-aggressive pilot who tends to fly his aircraft at its absolute limit, but lacks the skill to do so all of the time. He occasionally exceeds the aerodynamic and structural limits of his airplane, resulting in less than maximum performance and less than ideal decision making.

At the other end of the spectrum is the pilot who lacks experience and confidence. He assumes his skills are inadequate for the situation at hand and is overly conservative in his decision making. He is too timid to fly the aircraft near its limits and so fails to achieve maximum performance when he needs it.

These are the two extremes in pilot skill level that are found in most tactical fighter units. Individuals who exhibit these extremes are rare, but so are the ideal pilots. Most pilots would perform at an intermediate level of pilot skill. They would fall slightly left or right of the ideal pilot on a distribution curve. See Figure 22. This distribution is based upon the experience of veteran air-to-air pilots who have observed the many levels of skill which tactical pilots possess. The pilot skill factor of -1 is assigned to the under-experienced pilot, 0 to the ideal pilot, and +1 to the over-confident pilot. The possibility

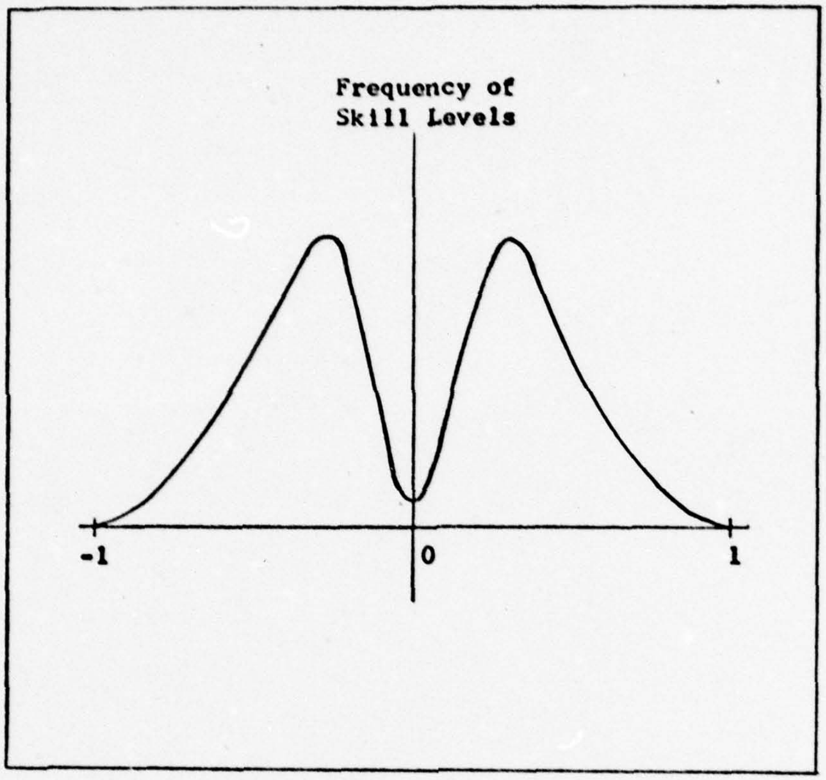


Fig. 22. Distribution of Pilot Skill Levels (Ref 6:37)

of varying the skill factor provides the facility to change the Smart Target level of capability.

Table II summarizes the extreme values of each of the decision parameters, as well as the nominal values selected for the ideal pilot. These values are provided to the program from SUBROUTINE SKILL when the Pilot Skill Factor is specified. See Appendix A for a detailed listing of SUBROUTINE SKILL.

Table II
Role Decision Parameters Based upon Pilot Skill Factor
as Determined by SUBROUTINE SKILL

Parameter	Pilot Skill Factor		
	-1	0	+1
ϕ_{D_2}	60^o	90^o	120^o
α_{max}*	15^o	30^o	40^o
θ_{D_4}	0^o	30^o	60^o
θ_{D_5}	30^o	60^o	90^o
ϵ_{D_6}	30^o	20^o	10^o
R_{max_T}	8,000 feet	6,000 feet	4,000 feet
R_{max_W}	30,000 feet	20,000 feet	10,000 feet
ϵ_{D_9}	0^o	10^o	20^o

*The values of α_{max} assume an α_{max} of the airplane of 30^o.

VI. Control Filter

In order that the Smart Target simulation be realistic, it is necessary to provide finite control rates to the target. This is accomplished through SUBROUTINE CONTRL. The control variables for the aircraft model are angle of attack, α, side slip angle, β, bank angle, ϕ, and thrust.

The inputs to this routine are the desired and actual values of the control variables along with the integration interval, Δt. This routine computes the new angular control rates $\dot{\alpha}$, $\dot{\beta}$, $\dot{\phi}$, and thrust rate. The actual values of the control variables are then calculated during the execution of CONTRL by a simple integration process.

Since the maximum rotational rate of the bank angle, ϕ, is such a strong function of α, $\dot{\phi}_{max}$ is computed.

The function $\dot{\phi}_{max}$ is a parabola fit through the following points:

α	$\dot{\phi}_{max}$
0°	180°/sec
10°	90°/sec
30°	24°/sec

Since the angles in the program are handled in radians,

$$\dot{\phi}_{max} = \pi - 10.9\,\alpha + 10.8862\,\alpha^2 \tag{21}$$

where $\dot{\phi}_{max}$ and α are expressed in radians/second and radians, respectively.

Once $\dot{\phi}_{max}$ is computed, the computation of the control variables is completed utilizing the scheme as shown in Fig. 23. The control α_X is used as an example.

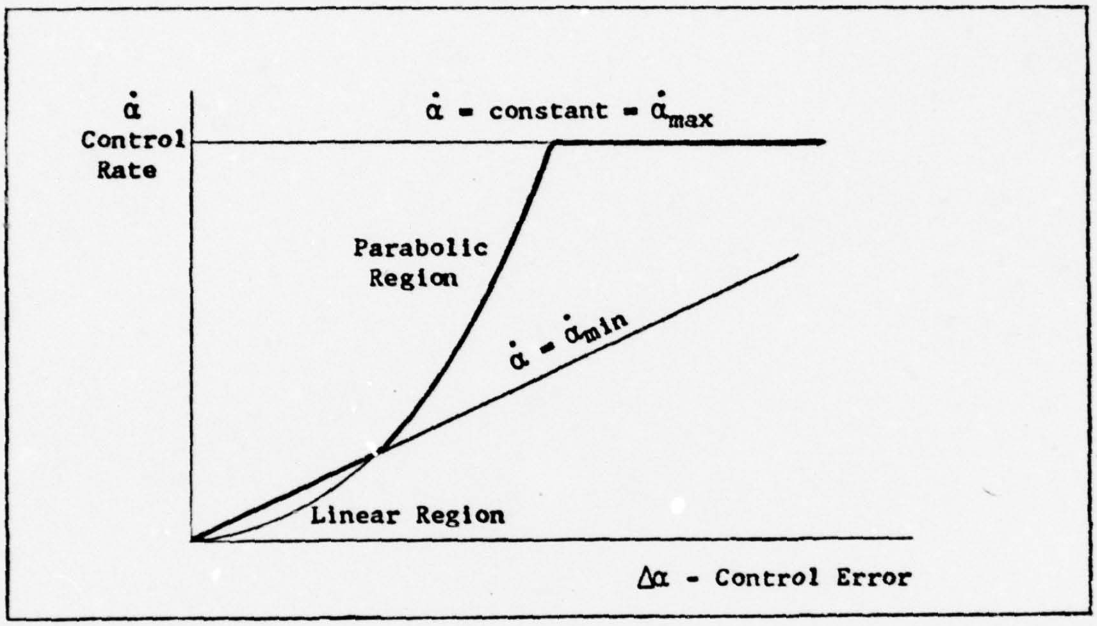


Fig. 23. Control Rate Model (Ref 4:94)

The relationship between the desired control error and the actual control rate is divided into three regions: linear, parabolic, and constant. The constant region represents the aircraft limit on the control error. The control error is given by

$$\Delta\alpha = \alpha_D - \alpha_X \tag{22}$$

where, $\Delta\alpha$ is the control error, α_D is the desired angle of attack, and α_X is the actual angle of attack.

Next, the proposed parabolic control rate $\dot{\alpha}$ is computed based on the maximum value of $\dot{\alpha}$. This maximum value is a function of the target aircraft being simulated.

$$\dot{\alpha} = \dot{\alpha}_{max} \left(\frac{4 \Delta\alpha}{\dot{\alpha}_{max}} \right)^2 \tag{23}$$

The proposed $\dot{\alpha}$ is examined to see if it falls outside the parabolic region. If

$$\dot{\alpha} \leq |\Delta\alpha| \tag{24}$$

then

$$\dot{\alpha} = \left| \frac{\Delta\alpha}{2 \, \Delta t} \right| \tag{25}$$

or, if,

$$\dot{\alpha} \geq \dot{\alpha}_{max} \tag{26}$$

then

$$\dot{\alpha} = \dot{\alpha}_{max} \tag{27}$$

The new value of the control variable is now computed

$$\alpha_{x_{new}} = \alpha_{x_{old}} + \mathrm{SIGN}(\dot{\alpha} \, \Delta t, \Delta\alpha) \tag{28}$$

where the function SIGN assigns to the absolute value of the first parameter, the sign of the second parameter.

A similar set of equations is used for each of the other controls. See Appendix A, SUBROUTINE CONTRL, for a detailed listing.

VII. Aircraft Model

Aircraft Equations of Motion

The differential equations of motion utilized in this simulation are for a rigid vehicle written in a combination of the wind and body axes (Ref 8:149-150). The assumptions of a flat earth, constant mass, and constant gravity are incorporated. The equations are:

$$T_{x_W} - D - mg \sin \theta_W = m \dot{V} \tag{29}$$

$$T_{y_W} - Y + mg \sin \phi_W \cos \theta_W = m V r_W \tag{30}$$

$$T_{z_W} - L + mg \cos \phi_W \cos \theta_W = -m V q_W \tag{31}$$

$$\dot{p} = \frac{1}{I_{xx}} \left[L + I_{xz} (\dot{r} + pq) + (I_{yy} - I_{zz}) qr \right] \tag{32}$$

$$\dot{q} = \frac{1}{I_{yy}} \left[M + I_{xz} (r^2 - p^2) + (I_{zz} - I_{xx}) rp \right] \tag{33}$$

$$\dot{r} = \frac{1}{I_{zz}} \left[N + I_{xz} (\dot{p} - qr) + (I_{xx} - I_{yy}) pq \right] \tag{34}$$

$$\dot{\theta}_W = q_W \cos \phi_W - r_W \sin \phi_W \tag{35}$$

$$\dot{\psi}_W = (q_W \sin \phi_W + r_W \cos \phi_W) \sec \theta_W \tag{36}$$

$$\dot{\phi}_W = p_W + q_W \sin \phi_W \tan \theta_W + r_W \cos \phi_W \tan \theta_W \tag{37}$$

$$\dot{x}_E = V \cos \theta_W \cos \psi_W \tag{38}$$

$$\dot{y}_E = V \cos \theta_W \sin \psi_W \tag{39}$$

$$\dot{z}_E = -V \sin \theta_W \tag{40}$$

Equations (29), (30), and (31) are the force equations written in the wind axes, where

T_{x_W} = the x component of the thrust vector.

T_{y_W} = the y component of the thrust vector.

T_{z_W} = the z component of the thrust vector.

D = the drag force on the aircraft.

Y = the side force on the aircraft.

L = the lift force on the aircraft.

Equations (32), (33), and (34) are the moment equations written in the body axes, where

p = rotational rate about the x body axis.

q = rotational rate about the y body axis.

r = rotational rate about the z body axis.

I_{xx} = principle moment of inertia about the x axis.

I_{yy} = principle moment of inertia about the y axis.

I_{zz} = principle moment of inertia about the z axis.

I_{xz} = product of inertia about the x-z plane.

L = rolling moment

M = pitching moment

N = yawing moment.

Although L is used both for lift force and rolling moment, the context makes it clear which is meant.

Equations (35), (36), and (37) are the Euler angle rates for the wind axes. Equations (38), (39), and (40) are the x, y, and z velocities which are integrated to obtain the position of the aircraft.

The control variables α, β, and ϕ_W are specified by the appropriate maneuver being simulated, and will not be described by the differential equations. In addition, since α and β are controlled, the relationship

between the wind and body axes may be determined. By taking the components of the angular velocity of F_B relative to F_W it follows that:

$$\begin{bmatrix} p_W \\ q_W \\ r_W \end{bmatrix} = L_{WB} \begin{bmatrix} p \\ q - \dot{\alpha} \\ r \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dot{\beta} \end{bmatrix} \tag{41}$$

where, L_{WB} is the transformation matrix

$$L_{WB} = \begin{bmatrix} c\,\alpha\,c\,\beta & s\,\beta & s\,\alpha\,c\,\beta \\ -c\,\alpha\,s\,\beta & c\,\beta & -s\,\alpha\,s\,\beta \\ -\,s\,\alpha & 0 & c\,\alpha \end{bmatrix} \tag{42}$$

The resulting scalar equations are:

$$p_W = p\cos\alpha\cos\beta + (q - \dot{\alpha})\sin\beta + r\sin\alpha\cos\beta \tag{43}$$

$$q_W = -p\cos\alpha\sin\beta + (q - \dot{\alpha})\cos\beta - r\sin\alpha\sin\beta \tag{44}$$

$$r_W = -p\sin\alpha + r\cos\alpha + \dot{\beta} \tag{45}$$

Solving Eq (44) for $\dot{\alpha}$ and Eq (45) for $\dot{\beta}$ gives:

$$\dot{\alpha} = q - q_W\sec\beta - p\cos\alpha\tan\beta - r\sin\alpha\tan\beta \tag{46}$$

$$\dot{\beta} = r_W + p\sin\alpha - r\cos\alpha \tag{47}$$

The quantities r_W and q_W are determined from Eqs (30) and (31), respectively.

$$r_W = \frac{1}{mV}\left(T_{y_W} - C + mg\sin\phi_W\cos\theta_W\right) \tag{48}$$

$$q_W = \frac{-1}{mV}\left(T_{z_W} - L + mg\cos\phi_W\cos\theta_W\right) \tag{49}$$

Now that the values of p_W, q_W, r_W, α, β, $\dot{\alpha}$, and $\dot{\beta}$ are known, the values of p, q, and r are determined.

$$\begin{bmatrix} p \\ q-\dot{\alpha} \\ r \end{bmatrix} = L_{BW} \begin{bmatrix} p_W \\ q_W \\ r_W \end{bmatrix} - L_{BW} \begin{bmatrix} 0 \\ 0 \\ \dot{\beta} \end{bmatrix} \tag{50}$$

where, L_{BW} is the transformation matrix

$$L_{BW} = \begin{bmatrix} c\,\alpha\, c\,\beta & -c\,\alpha\, s\,\beta & -s\,\alpha \\ s\,\beta & c\,\beta & 0 \\ s\,\alpha\, c\,\beta & -s\,\alpha\, s\,\beta & c\,\alpha \end{bmatrix} \tag{51}$$

The scalar quantities p, q, and r follow:

$$p = p_W \cos\alpha \cos\beta - q_W \cos\alpha \sin\beta - (r_W - \dot{\beta}) \sin\alpha \tag{52}$$

$$q = \dot{\alpha} + p_W \sin\beta + q_W \cos\beta \tag{53}$$

$$r = p_W \sin\alpha \cos\beta - q_W \sin\alpha \sin\beta + (r_W - \dot{\beta}) \cos\alpha \tag{54}$$

Since the controls provided to the target are for a trimmed condition, it is assumed that actual control deflections of elevator, aileron, and rudder are provided such that the moment equations yield the same values of p, q, and r as Eqs (52), (53), and (54). For this reason, the moment equations are not utilized in this program.

In summary, the actual equations of motion for the Smart Target are the following:

$$\dot{V} = \frac{T_{x_W} - D}{M} - g \sin\theta_W \tag{55}$$

$$\dot{\theta}_W = q_W \cos \phi_W - r_W \sin \phi_W \quad (56)$$

$$\dot{\psi}_W = (q_W \sin \phi_W + r_W \cos \phi_W) \sec \theta_W \quad (57)$$

$$\dot{x} = V \cos \theta_W \cos \psi_W \quad (58)$$

$$\dot{y} = V \cos \theta_W \sin \psi_W \quad (59)$$

$$\dot{z} = -V \sin \theta_W \quad (60)$$

where r_W is given by Eq (48), q_W is given by Eq (49), and p, q, and r are determined from Eqs (52), (53), and (54), respectively.

Eqs (55) through (60) are integrated in SUBROUTINE RKDES each time through the program to update the flight path of the target. Refer to Appendix A for a listing of this subroutine.

F-4 Target Model

The F-4 model used in Smart Target was developed by the National Aeronautics and Space Administration (NASA) for their simulation studies. The aircraft used throughout this study was programmed to be representative of the F-4E because of the author's flying experience and knowledge of this aircraft. This evaluation does not cover the entire operational envelope of the F-4 aircraft, but concentrates on the region usually encountered in air-to-air combat, namely, mach numbers below 1.2 and altitudes below 35,000 feet.

Thrust parameters for the model are stored in tabular form as functions of Mach Number. From these thrust parameters, the thrust available is computed as a function of altitude. The target aerodynamics are specified by various stability derivatives which are stored

as functions of Mach Number and angle of attack. Since the target is assumed to be a trimmed model, the stability derivatives with respect to control deflections are assumed to be zero. The stability derivatives, thrust parameters, and equations for the aerodynamic coefficients used in this program appear in Appendix B.

Opponent Model

In actual implementation of the Smart Target Program, the LAMARS simulator will provide the opponent's inputs. However, in this study there was no requirement for the opponent to be responsive and a simple model for the opponent was needed to test the target simulation.

The state of the opponent is given in the form of nine equations with each of the state variables as functions of time. The nine components of the opponent's state are:

x_{OPP}, y_{OPP}, z_{OPP}	components of the opponent's position
u_{OPP}, v_{OPP}, w_{OPP}	components of the opponent's velocity vector
ψ_{FE}, ϕ_{FE}, θ_{FE}	opponent's Euler angles.

The state equations must be changed for each maneuver flown by the opponent.

In order to evaluate the maneuvers flown by the Smart Target, two flight paths were programmed for the opponent. In the first case, the opponent was programmed to remain in a straight and level condition, as would be the case if he did not see the target. The opponent's

state equations are:

$$x_{OPP} = x_{OPP} + u_{OPP}\,\Delta t \tag{61}$$

$$y_{OPP} = y_{OPP} + v_{OPP}\,\Delta t \tag{62}$$

$$z_{OPP} = z_{OPP} + w_{OPP}\,\Delta t \tag{63}$$

$$u_{OPP} = 800 \tag{64}$$

$$v_{OPP} = 0 \tag{65}$$

$$w_{OPP} = 0 \tag{66}$$

$$\psi_{FE} = 0 \tag{67}$$

$$\phi_{FE} = 0 \tag{68}$$

$$\theta_{FE} = 0 \tag{69}$$

The second maneuver simulated for the opponent is a hard turn. This maneuver was used as a defensive maneuver to evaluate the target as an attacker. The opponent's state equations for x_{OPP}, y_{OPP}, and z_{OPP} remain the same. The other state equations are:

$$u_{OPP} = V_{TOPP} \cos \psi_{FE} \tag{70}$$

$$v_{OPP} = V_{TOPP} \sin \psi_{FE} \tag{71}$$

$$w_{OPP} = 0 \tag{72}$$

$$\psi_{FE} = \psi_{FE} + \frac{\pi}{18}\Delta t \tag{73}$$

$$\phi_{FE} = \frac{\pi}{2} \tag{74}$$

$$\theta_{FE} = 0 \tag{75}$$

VIII. Additional Subroutines and Function Subprograms

In addition to the main Smart Target routines already described, there are several other subroutines and function subprograms which are essential in completing this simulation. A brief description of these subprograms is listed below.

SUBROUTINE TRNSSB

This subroutine transforms a vector in the body fixed reference frame, F_B, into the stability axis system, F_S. It uses an Euler angle transformation through the angle of attack, α. It is called from the main program, PROGRAM SMTTGT.

SUBROUTINE ATMOS

This routine uses the two lowest layers of the 1962 United States Standard Atmosphere model to determine the air density, speed of sound, and density ratio. The input required is the aircraft altitude in feet.

FUNCTION EXTRA

This function accomplishes a two-dimensional interpolation of the aerodynamic coefficients and stability derivatives. All coefficient tables are input as a function of angle of attack and Mach number, except for thrust parameters which are a function of altitude and Mach number.

SUBROUTINE EULER

This subroutine is used to solve the six differential equations of motion. The integration technique is a straightforward Euler method.

SUBROUTINE ISELCT

This routine selects one of six variables on a random basis. In evasive and defensive situations ISELCT is called by FUNCTION ISTATE to randomly select a maneuver. The random number used in this selection is obtained by calling SUBROUTINE RANDU.

SUBROUTINE RANDU

This subroutine utilizes several library subprograms of the CDC 6600 digital computer. These subprograms, RANSET, RANF, and RANGET, are used to randomly select an integer and a floating point random number for use in SUBROUTINE DESIRE and SUBROUTINE ISTATE.

Listings of these additional subroutines and function subprograms are found in Appendix A.

IX. Results

Once the necessary modifications were made so that the Smart Target Program would be compatible with the CDC 6600 computer, the F-4 aircraft model was attached. The first step in evaluating the simulation was to determine if the initial situation cell and corresponding combat maneuver were selected properly. There were 37 sets of initial conditions, each corresponding to a situation cell, which were used to test this objective. Examination of the results indicated that the situation cell and maneuver, called for in each case, were exactly as expected for the ideal pilot with a skill factor of zero. It was also found that the situation cells and tactics decisions were altered accordingly for the pilots with skill factors of +1 and -1.

The second step in this evaluation was to determine if the target was actually maneuvering in response to his opponent with the desired controls to perform each of the selected air combat maneuvers. It was found that the desired control inputs to generate each maneuver, as programmed by Barrett (Ref 6), were extremely inadequate. In most cases, the target did not maneuver in response to his opponent, and the actual maneuver flight paths were unrealistic. One major deficiency noted in the evasive and defensive roles was that because each maneuver is selected on a random basis, there was never sufficient time allowed to partially complete these maneuvers. As a result, the target would switch randomly between evasive or defensive maneuvers, and never establish a consistent flight path long enough to be an effective maneuver. An extensive modification was accomplished to SUBROUTINE DESIRE, in an

attempt to correct these problems, so that the target was correctly responding and generating realistic maneuver flight paths. With the incorporated modifications, the execution of the defensive and evasive situations proved satisfactory. The maneuvers in the offensive and attack role also proved adequate on an individual basis, but when the maneuvers were selected in a running battle, it was not conclusively shown that the target was correctly responding to the opponent. Contained in Appendix C are the results obtained in simulating each Air Combat Maneuver.

Four test cases were run to ensure that the target was responding correctly and continually updating its maneuvers based on its current situation cell. Within each case the pilot skill factor was set at -1, -.5, 0, .5, and 1, to check the effect of the pilot on each selected maneuver. The initial conditions for each of these test cases are listed in Table III.

Test Case 1 was set up so that the ideal target would be evasive. The initial situation cell number was 202 and the maneuver selected was the high g roll over. After 16 seconds, the target entered the offensive role and selected the low speed yo-yo to increase his overtake on the opponent. As the target continued to close on the opponent, pure pursuit and the low speed yo-yo were alternately selected until the completion of the run at 60 seconds. Figure 24 shows the three-dimensional flight paths and two-dimensional ground track of the ideal target and opponent in the evasive test case.

The second test case was initially programmed for the ideal target to be defensive. The hard turn was selected as the first maneuver, with the situation cell being number 301. The situation cell changed after

Table III
Initial Conditions for Smart Target Test Runs

Variable	Case 1	Case 2	Case 3	Case 4
x_{TE} (ft)	1,700	13,000	-8,000	-5,000
y_{TE} (ft)	0	0	0	0
z_{TE} (ft)	-20,000	-20,000	-20,000	-20,000
u_{TE} (ft/sec)	800	800	655	800
v_{TE} (ft/sec)	0	0	459	0
w_{TE} (ft/sec)	0	0	0	0
ψ_{TE} (deg)	0	0	35	0
θ_{TE} (deg)	0	0	0	0
ϕ_{TE} (deg)	0	0	0	0
p_T (deg/sec)	0	0	0	0
q_T (deg/sec)	0	0	0	0
r_T (deg/sec)	0	0	0	0
x_{OPP} (ft)	0	0	0	0
y_{OPP} (ft)	0	0	0	0
z_{OPP} (ft)	-20,000	-20,000	-20,000	-20,000
u_{OPP} (ft/sec)	900	900	900	800
v_{OPP} (ft/sec)	0	0	0	0
w_{OPP} (ft/sec)	0	0	0	0
ψ_{FE} (deg)	0	0	0	0
θ_{FE} (deg)	0	0	0	0
ϕ_{FE} (deg)	0	0	90	90

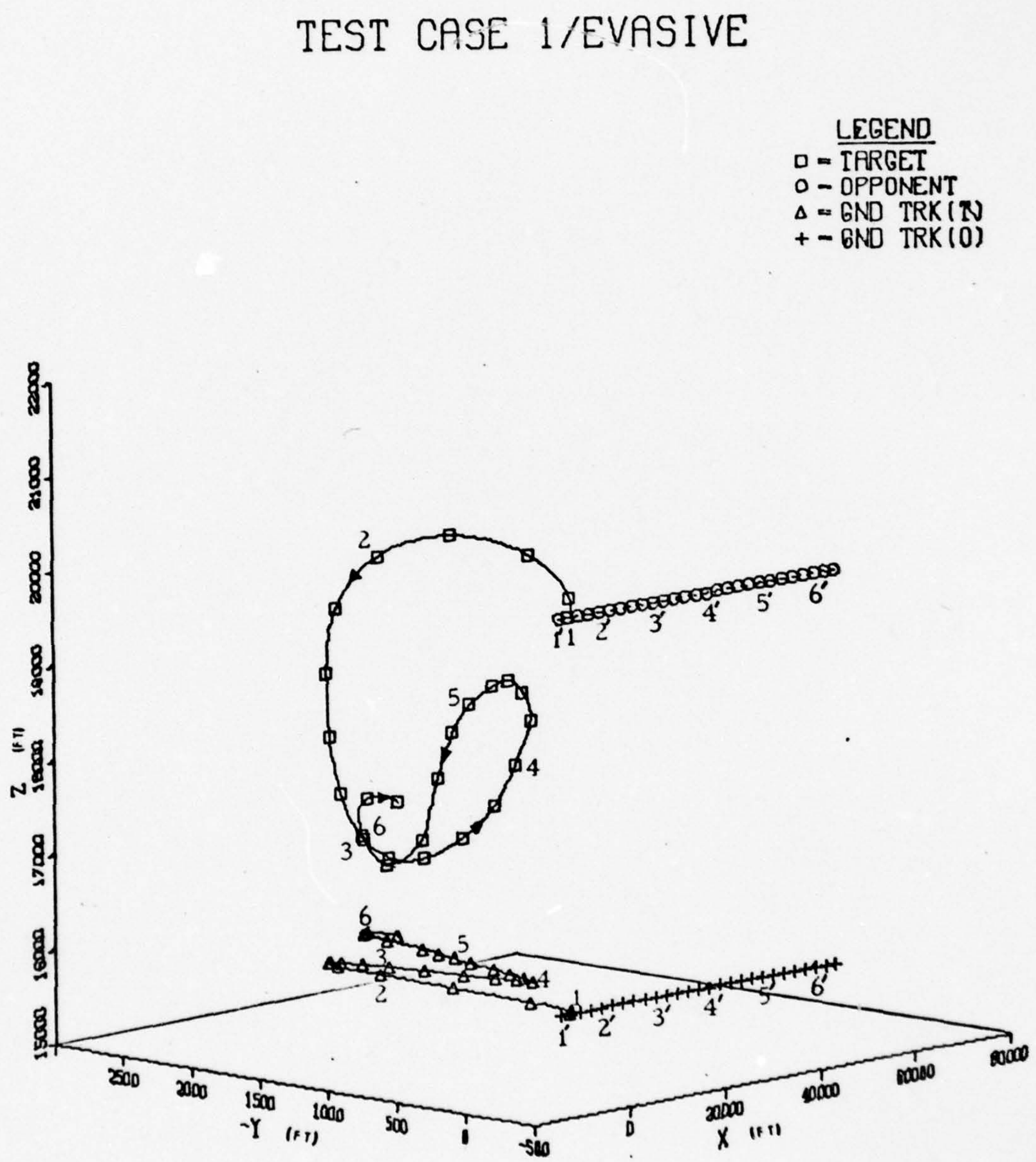


Fig. 24. Responsive Simulation in the Evasive Role

12.5 seconds, and the target selected the maximum energy maneuver. As the target gained velocity and entered the opponent's rear hemisphere, the low speed yo-yo was chosen in order to close on the opponent. This took place after 22.5 seconds into the simulation. As the target continued to decrease range, pure pursuit was selected after 40.5 seconds. The target remained in pursuit until the completion of the test run. See Fig. 25 for the plot of the defensive test case.

Test Case 3 was initialized so that the ideal target would be in the offensive role. The first maneuver selected was the low speed yo-yo to close on the opponent. The situation cell was number 403. After 5.0 seconds into the run, the target selected pure pursuit. As the target turned with the opponent, the rate of closure decreased and the low speed yo-yo was selected to regain airspeed at 10.5 seconds. At 15.5 seconds the target chose lag pursuit and continued to track the opponent until 23.0 seconds. At this point the situation cell called for a barrel roll attack. The target continued in the offensive role, but was unable to improve his situation with the selection of the barrel roll. The barrel roll was flown until the end of the test run at 60.0 seconds. Fig. 26 shows this offensive test run.

In the fourth test case, the ideal target was set up in the attack role. The initial situation cell was 504. The maneuver selected was the pure pursuit curve for a missile attack. The target tracked the opponent for 7.5 seconds, but was unable to null the steering error to achieve a kill. At this point the low speed yo-yo was selected, and the target attempted to close on the opponent. After 36.5 seconds, the target regained the attack role and selected the lead pursuit curve. While still in the attack role at 42.0 seconds, the barrel roll attack was

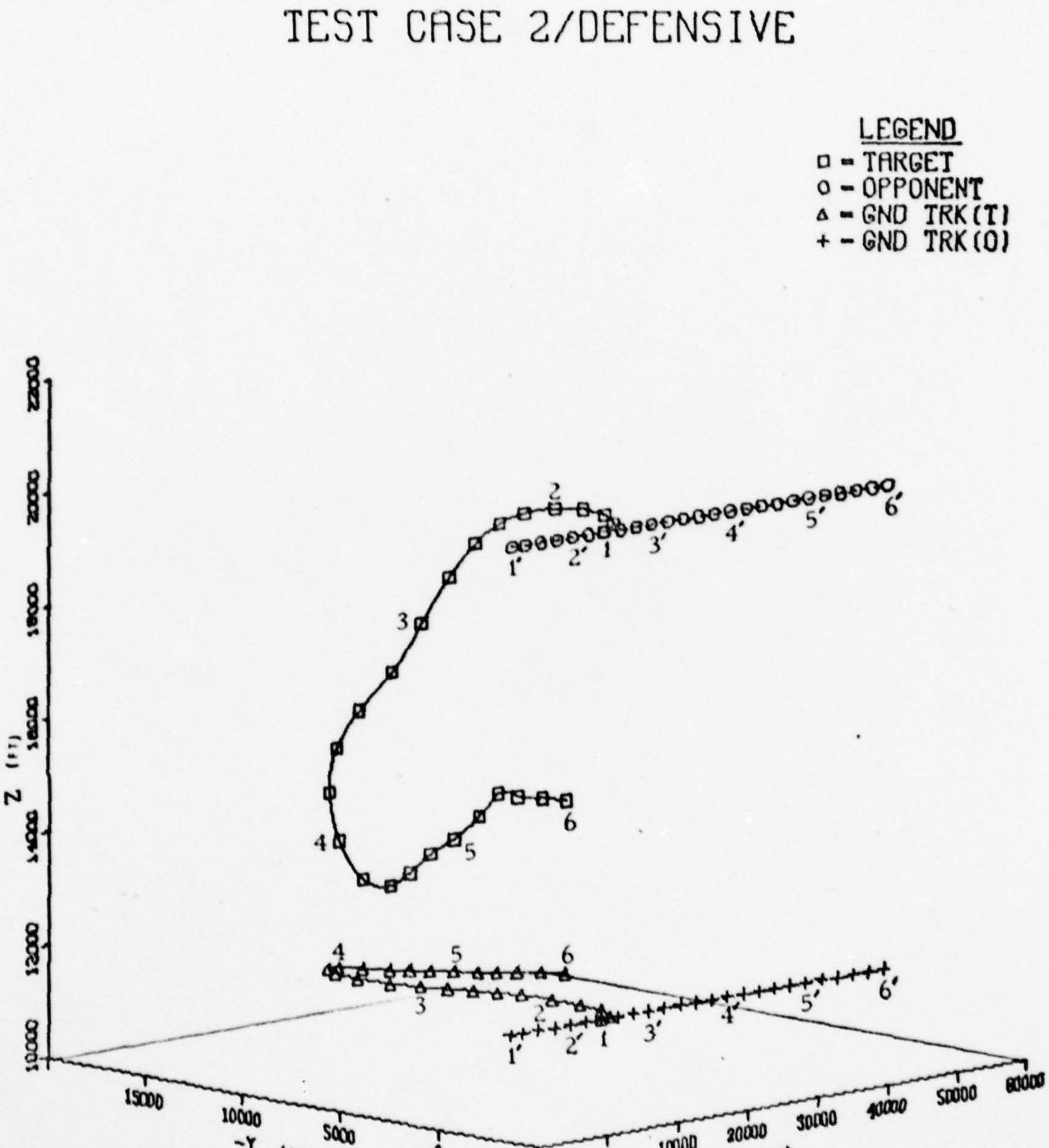


Fig. 25. Responsive Simulation in the Defensive Role

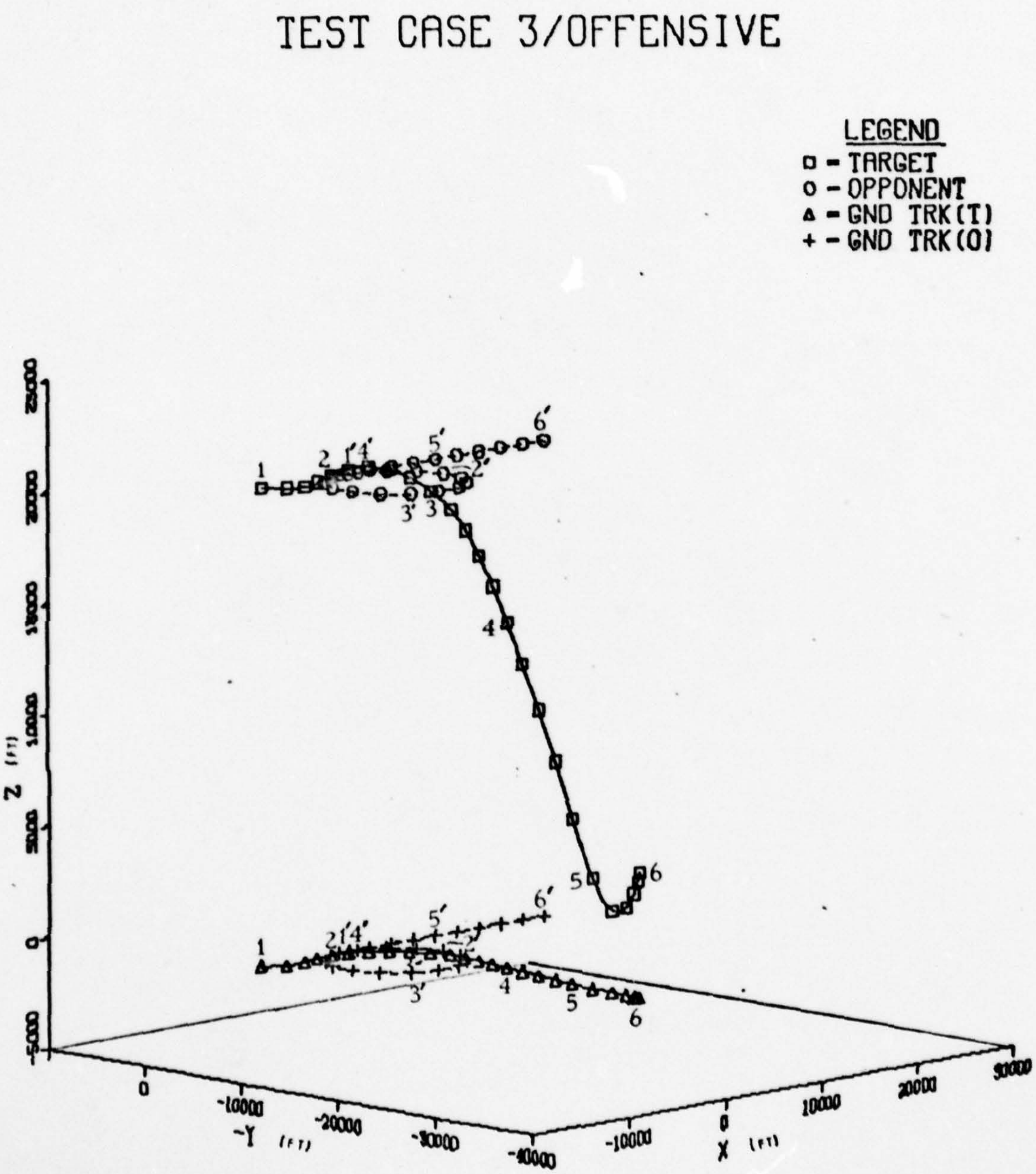


Fig. 26. Responsive Simulation in the Offensive Role

chosen. After 46.5 seconds, the low speed yo-yo was again selected and maintained until the end of the run at 60.0 seconds. The attack test results are shown in Fig. 27.

The final objective of this study was to evaluate the effect of the different pilot skill levels to determine the validity of the pilot model. Upon examination of the results obtained from the four test cases, using the five pilot skill levels mentioned above, the pilot model proved realistic throughout the simulation. For example, in Test Case 2, the ideal pilot was able to convert the defensive situation into an attack position. The timid pilot with skill level of -1.0, after the initial defensive maneuver, selected the maximum energy maneuver in an attempt to disengage. The over-aggressive pilot with skill level of 1.0, did convert to the offensive role. However, because the control inputs utilized were excessive, the over-confident pilot lost valuable airspeed and maneuvering potential and could not close sufficiently on the opponent to be a threat. As expected, the pilots with skill factors of 0.5 and -0.5 performed identically. Both were able to convert from the defensive to the attack role. However, it took a longer period of time for these two pilots to make this conversion than did the ideal pilot.

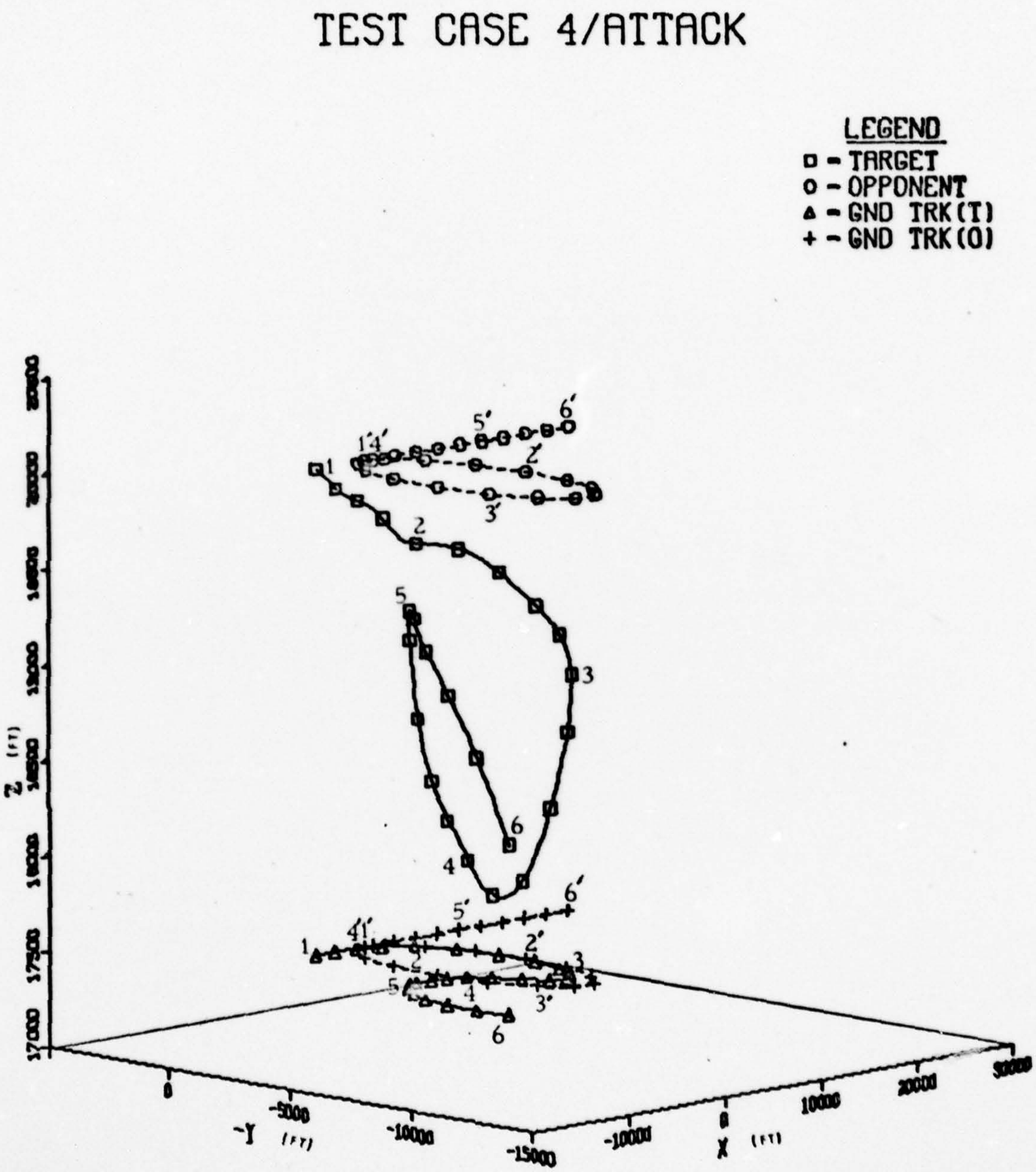


Fig. 27. Responsive Simulation in the Attack Role

X. Conclusions and Recommendations

Conclusions

The objectives of this study, as listed in Chapter I, have all been accomplished to varying degrees.

The selection of maneuvers based on the situation space defined, and the factors utilized to make the selection, were all accomplished satisfactorily. However, the decisions used to define this situation space and maneuver selection were based only on the experience of veteran F-4 pilots. With the advent of high thrust-to-weight ratio fighters, with greater turning and acceleration capability, the situation space will need to be modified to account for these higher performance aircraft. In addition, with the ever increasing capability of air-to-air weapons, further modifications in the situation space and decision parameters must be incorporated for the Smart Target Simulation to be effective.

The response of the target to the opponent's maneuvering and the ability of the target to realistically fly each maneuver have not been totally solved to the author's satisfaction. Although the controls to fly each maneuver have proved sufficient in this study, actual implementation of this program on the LAMARS may require additional modification to the DESIRE subroutine. Because the offensive and attack maneuvers are so dependent on the relative states of the two aircraft, and because the same maneuver may be selected for several situation cells, the present set of controls may not prove adequate for all selected situations. This is especially true regarding the barrel roll attack. Also, the

simple unresponsive opponent model used in this study definitely does not provide a true test for the offensive and attack maneuvers. Actual evaluation of these roles may only prove satisfactory after the actual LAMARS simulation is accomplished. It is also concluded that there may be other weaknesses in the control laws which may be found and altered after use. In addition, SUBROUTINE DESIRE must be altered for each simulated target other than the F-4.

The target has shown that it does respond continuously to the opponent's maneuvers by changing its maneuver in response to the opponent. It has also shown that the variation in pilot skill has a definite affect on the decisions reached and the maneuvers selected.

Recommendations

Further development of the air combat maneuvers should be anticipated. As new combat tactics are developed with the changing fighter and weapons capabilities, new maneuvers and changes to existing maneuvers must be continually updated.

With the implementation of this program on the LAMARS System, further development of the basic maneuvers, in response to an actual responsive opponent, needs to be incorporated. In addition, a more advanced integration routine may be employed if the Euler technique is not sufficiently accurate for smaller increments of time.

Applications

There are many applications foreseen in using this responsive target simulation. First of all, the savings in training costs would be significant. One simulator sortie would accomplish the same amount

of training as two or more actual flying sorties. Also, the matter of flying safety in the dangerous air combat mission would be alleviated.

Probably the greatest application of this type of simulation would be to actually program Soviet aircraft vehicle dynamics and maneuvers so that realistic dogfights could be portrayed in a training environment. This would enable our combat crews to be even better prepared when facing the enemy in a combat environment.

In addition, a responsive model such as Smart Target, would be a valuable tool in simulator evaluations of new air-to-air technology. For example, air-to-air gun sights, air-to-air weapons, and even new design aircraft could be tested even before actual development and production of such systems.

Bibliography

1. Greene, Terrill E. and John H. Huntzicker. A Survey of Methods for Studying Air-to-Air Combat: World War II to the Present. RM-5351/1-PR. Santa Monica, California: The Rand Corporation, December 1967.

2. Hutcheson, J. H. and R. Segerblom. TACTICS: A Three-Body, Three-Dimensional Intercept Simulation Program. RM-5759-PR. Santa Monica, California: The Rand Corporation, October 1969.

3. Welch, Larry D. and John L. Pickett. TAC AVENGER - Conception to Maturity. Volume II. AFFDL-TR-72-57. Wright-Patterson AFB, Ohio: Air Force Flight Dynamics Laboratory, May 1972.

4. Hague, D. S., et al. Combat Optimization and Analysis Program--COAP. Volumes I-IV. AFFDL-TR-71-52. Wright-Patterson AFB, Ohio: Air Force Flight Dynamics Laboratory, August 1971.

5. Burgin, G. H., et al. An Adaptive Maneuvering Logic Program for the Simulation of One-on-One Air-to-Air Combat. Volumes I-II. NASA CR-2582. San Diego, California: Decision Science, Incorporated, September 1975.

6. Barrett, Robert P. Development of a Responsive Target for Tracking Tasks in the Dynamic Environmental Simulator. Master's Thesis. Wright-Patterson AFB, Ohio: Air Force Institute of Technology, June 1973.

7. Kuhns, James, et al. Research on "Smart Target" for Aerial Combat. Volumes I-II. AMRL-TR-75-83. McLean, Virginia: Quest Research Corporation, August 1976.

8. Etkin, Bernard. Dynamics of Atmospheric Flight. New York, New York: John Wiley & Sons, Inc., 1972.

9.

10.

11. Mayer, Lester G., et al. Study/Research for Air-to-Air Combat Simulator. Volume III. ASD-TR-68-42. Wright-Patterson AFB, Ohio: Aeronautical Systems Division, September 1968.

12. -----. Fortran Extended Version 4 Reference Manual. Control Data Corporation, Sunnyvale, California, 1975.

13. -----. _Digital Computation Handbook of Available Subroutines for the Control Data 6600 Computer Systems_. ASD/VNC Internal Memo 70-003. Wright-Patterson AFB, Ohio: Aeronautical Systems Division, 15 December 1970.

Appendix A

Smart Target Computer Program Listing

This appendix contains the Fortran program listings of each of the programs, subroutines, and function subprograms used in this Smart Target simulation study.

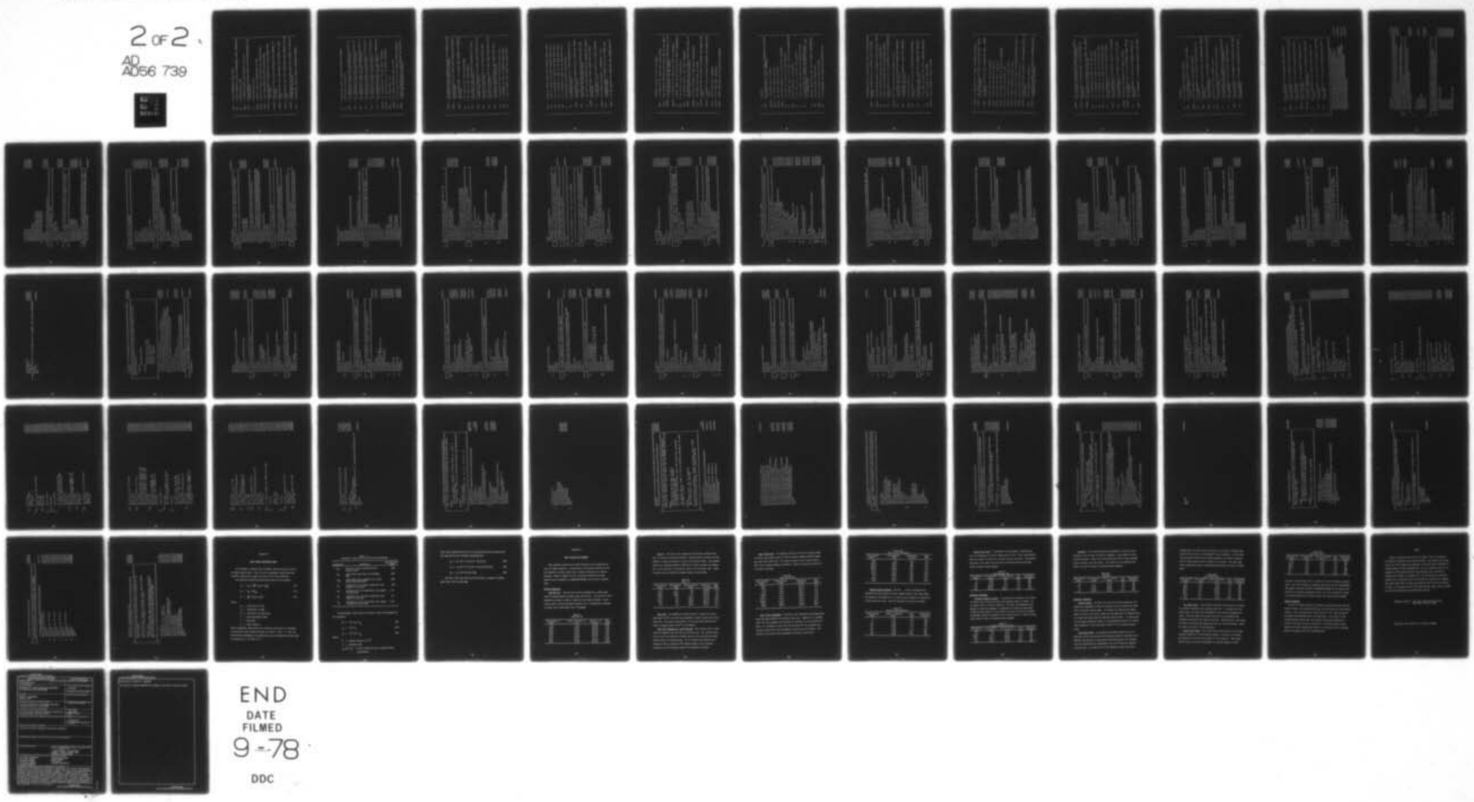
```
      PROGRAM SMTTGT(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE3)
C*********************************************************************
C* THIS PROGRAM SIMULATES SMART TARGET MANEUVERS AGAINST A FIGHTER   *
C*  OPPONENT                                                         *
C*                                                                   *
C* THE FOLLOWING VARIABLES ARE USED IN THIS SIMULATION:              *
C*                                                                   *
C*     A       - ANGLE OF ATTACK USED IN THE AERODYNAMIC TABLES, USED IN *
C*               CONVERTING AERODYNAMIC TABLES TO BODY AXES FROM     *
C*               STABILITY AXES.                                     *
C*                                                                   *
C*     A1      - AN INTERMEDIATE VARIABLE EQUAL TO F37(I,J) IN CONVERSION*
C*               OF AERODYNAMIC TABLES.                              *
C*                                                                   *
C*     A2      - AN INTERMEDIATE VARIABLE EQUAL TO F42(I,J) IN CONVERSION*
C*               OF AERODYNAMIC TABLES.                              *
C*                                                                   *
C*     A3      - AN INTERMEDIATE VARIABLE EQUAL TO F38(I,J) IN CONVERSION*
C*               OF AERODYNAMIC TABLES.                              *
C*                                                                   *
C*     A4      - AN INTERMEDIATE VARIABLE EQUAL TO F43(I,J) IN CONVERSION*
C*               OF AERODYNAMIC TABLES.                              *
C*                                                                   *
C*     AINT    - AN INTERMEDIATE VARIABLE USED IN DETERMINATION OF   *
C*               ANGULAR RATES.                                      *
C*                                                                   *
C*     ALDTMX  - THE MAXIMUM VALUE ALLOWED FOR THE RATE OF CHANGE OF *
C*               ANGLE OF ATTACK.                                    *
C*                                                                   *
C*     ALIFT   - THE LIFT FORCE ON THE TARGET AIRCRAFT = CL*Q*S      *
C*                                                                   *
C*     ALPHAD  - THE DESIRED VALUE OF ANGLE OF ATTACK CALLED FOR IN  *
C*               SUBROUTINE DESIRE (RADIANS).                        *
C*                                                                   *
C*     ALPHAT  - THE ANGLE BETWEEN THE THE AIRCRAFT AXIS AND THE ENGINE *
```

```
*               THRUST VECTOR (RADIANS).                                     *
*                                                                            *
*      ALPHAX - THE ANGLE OF ATTACK OF THE TARGET AIRCRAFT (RADIANS).        *
*                                                                            *
*      ALPHDT - THE RATE OF CHANGE OF THE ANGLE OF ATTACK (RADIANS/SEC).     *
*                                                                            *
*      ALPHO  - ANGLE OF ATTACK IN DEGREES FOR OUTPUT PURPOSES.              *
*                                                                            *
*      ALPMAX - THE MAXIMUM VALUE THE TARGET ANGLE OF ATTACK CAN ASSUME      *
*               (RADIANS).                                                   *
*                                                                            *
*      ALPMIN - THE MINIMUM VALUE THE TARGET ANGLE OF ATTACK CAN ASSUME      *
*               (RADIANS).                                                   *
*                                                                            *
*      AMACHT - THE MACH NUMBER OF THE TARGET AIRCRAFT.                      *
*                                                                            *
*      AMASS  - THE MASS OF THE TARGET AIRCRAFT (SLUGS).                     *
*                                                                            *
*      ANGOFF - THE TARGET'S ANGLE OFF (DEGREES).                            *
*                                                                            *
*      ARCSIN - AN INTERMEDIATE VARIABLE USED IN THE DESIRE SUBROUTINE.      *
*                                                                            *
*      B      - WINGSPAN (FEET).                                             *
*                                                                            *
*      BEDTMX - THE MAXIMUM ALLOWABLE RATE OF CHANGE OF SIDESLIP ANGLE       *
*               (RADIANS/SEC).                                               *
*                                                                            *
*      BETA   - SIDESLIP ANGLE (RADIANS).                                    *
*                                                                            *
*      BETADT - RATE OF CHANGE OF SIDESLIP ANGLE (RADIANS/SEC).              *
*                                                                            *
*      BETAD  - THE DESIRED VALUE OF SIDESLIP ANGLE CALLED FOR IN            *
*               SUBROUTINE DESIRE (RADIANS).                                 *
*                                                                            *
*      BETMAX - THE MAXIMUM VALUE OF SIDESLIP ANGLE (RADIANS).               *
```

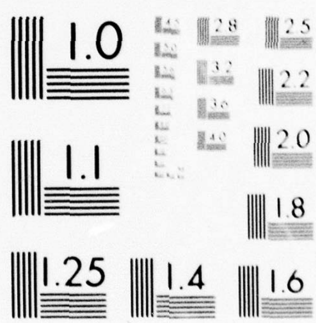
```
*                                                                      *
*   BETO    - SIDESLIP ANGLE IN DEGREES FOR OUTPUT PURPOSES.          *
*                                                                      *
*   BINT    - AN INTERMEDIATE VARIABLE USED IN THE COMPUTATION OF THE *
*             ANGULAR RATES.                                           *
*                                                                      *
*   C11     - DIRECTION COSINE FOR CONVERTING FROM THE EARTH TO THE    *
*             BODY AXES.                                               *
*                                                                      *
*   C12     - DIRECTION COSINE FOR CONVERTING FROM THE EARTH TO THE    *
*             BODY AXES.                                               *
*                                                                      *
*   C13     - DIRECTION COSINE FOR CONVERTING FROM THE EARTH TO THE    *
*             BODY AXES.                                               *
*                                                                      *
*   C21     - DIRECTION COSINE FOR CONVERTING FROM THE EARTH TO THE    *
*             BODY AXES.                                               *
*                                                                      *
*   C22     - DIRECTION COSINE FOR CONVERTING FROM THE EARTH TO THE    *
*             BODY AXES.                                               *
*                                                                      *
*   C23     - DIRECTION COSINE FOR CONVERTING FROM THE EARTH TO THE    *
*             BODY AXES.                                               *
*                                                                      *
*   C31     - DIRECTION COSINE FOR CONVERTING FROM THE EARTH TO THE    *
*             BODY AXES.                                               *
*                                                                      *
*   C32     - DIRECTION COSINE FOR CONVERTING FROM THE EARTH TO THE    *
*             BODY AXES.                                               *
*                                                                      *
*   C33     - DIRECTION COSINE FOR CONVERTING FROM THE EARTH TO THE    *
*             BODY AXES.                                               *
*                                                                      *
*   CA      - THE VELOCITY OF SOUND (FEET/SEC).                        *
*                                                                      *
```

AD-A056 739 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO SCH--ETC F/G 15/7
DEVELOPMENT OF SMART TARGET FOR SIMULATION OF ONE-ON-ONE AIR-TO--ETC(U)
MAR 78 D A LEUTHAUSER
UNCLASSIFIED AFIT/GAE/AA/78M-7 NL

2 OF 2
AD
A056 739



END
DATE
FILMED
9-78
DDC



```
*       CBAR    - MEAN AERODYNAMIC CHORD (FEET).                              *
*                                                                             *
*       CCA     - COSINE SQUARED (A) USED IN AERODYNAMIC TABLE CONVERSION.    *
*                                                                             *
*       CD      - COEFFICIENT OF DRAG.                                        *
*                                                                             *
*       CFAC    - A RANDOM VARIABLE WITH VALUE OF +1 OR -1.                   *
*                                                                             *
*       CINT    - AN INTERMEDIATE VARIABLE USED IN THE COMPUTATION OF THE     *
*                 ANGULAR RATES.                                              *
*                                                                             *
*       CL      - COEFFICIENT OF LIFT.                                        *
*                                                                             *
*       CLMAX   - MAXIMUM VALUE FOR COEFFICIENT OF LIFT.                      *
*                                                                             *
*       COEF2   - BASIC LIFT FORCE COEFFICIENT WITH ELEVATOR ZERO.            *
*                                                                             *
*       COEF3   - BASIC DRAG COEFFICIENT WITH ELEVATOR ZERO.                  *
*                                                                             *
*       COEF12  - DRAG COEFFICIENT INCREMENT DUE TO FULL SPEED BRAKE          *
*                 DEFLECTION.                                                 *
*                                                                             *
*       COEF39  - DERIVATIVE OF SIDE FORCE COEFFICIENT WITH RESPECT TO        *
*                 ROLL RATE.                                                  *
*                                                                             *
*       COEF41  - DERIVATIVE OF LIFT COEFFICIENT WITH RESPECT TO PITCH        *
*                 RATE.                                                       *
*                                                                             *
*       COEF44  - DERIVATIVE OF SIDE FORCE COEFFICIENT WITH RESPECT TO YAW    *
*                 RATE.                                                       *
*                                                                             *
*       COEF46  - DERIVATIVE OF LIFT COEFFICIENT WITH RESPECT TO ANGLE OF     *
*                 ATTACK RATE OF CHANGE.                                      *
*                                                                             *
*       CPH     - COS(PHITER), USED TO COMPUTE DIRECTION COSINES.             *
```

```
CTH    - COS(THETER), USED TO COMPUTE DIRECTION COSINES.

CY     - COEFFICIENT OF SIDEFORCE.

DELU   - THE DIFFERENCE BETWEEN THE X COMPONENTS OF VELOCITY OF
         THE OPPONENT AND TARGET AIRCRAFT (FEET/SEC).

DELV   - THE DIFFERENCE BETWEEN THE Y COMPONENTS OF VELOCITY OF
         THE OPPONENT AND TARGET AIRCRAFT (FEET/SEC).

DELW   - THE DIFFERENCE BETWEEN THE Z COMPONENTS OF VELOCITY OF
         THE OPPONENT AND TARGET AIRCRAFT (FEET/SEC).

DELX   - THE DIFFERENCE BETWEEN THE X COMPONENTS OF POSITION OF
         THE OPPONENT AND TARGET AIRCRAFT (FEET).

DELY   - THE DIFFERENCE BETWEEN THE Y COMPONENTS OF POSITION OF
         THE OPPONENT AND TARGET AIRCRAFT (FEET).

DELZ   - THE DIFFERENCE BETWEEN THE Z COMPONENTS OF POSITION OF
         THE OPPONENT AND TARGET AIRCRAFT (FEET).

DEVANG - THE DEVIATION ANGLE FOR THE TARGET (DEGREES).

DEVOPP - THE DEVIATION ANGLE OF THE OPPONENT (DEGREES).

DRAG   - THE DRAG FORCE (LBS.).

DYNPRS - THE DYNAMIC PRESSURE, 1/2*RHO *V**2.

EPSD   - THE LAG ANGLE USED FOR THE PURSUIT CURVES (RADIANS).

EPSD6  - MAXIMUM DEVIATION ANGLE OF THE OPPONENT FOR WHICH THE
         TARGET WILL BE EVASIVE (DEGREES).
```

```
EPS09   - THE DEVIATION ANGLE ADVANTAGE ALLOWED THE OPPONENT WHILE
          REMAINING ON THE OFFENSIVE (DEGREES).

ETADF   - THE ELEVATION RATE THE TARGET SEES VIEWING THE OPPONENT
          (RADIANS/SEC).

ETAFR   - THE ELEVATION ANGLE THE TARGET SEES VIEWING THE OPPONENT
          (RADIANS).

F2      - THE TABLE OF LIFT FORCE COEFFICIENTS.

F3      - THE TABLE OF DRAG COEFFICIENTS.

F12     - THE TABLE OF DRAG COEFFICIENT INCREMENTS DUE TO FULL
          SPEED BRAKE DEFLECTION.

F39     - THE TABLE OF DERIVATIVES OF SIDE FORCE COEFFICIENT WITH
          RESPECT TO ROLL RATE.

F41     - THE TABLE OF DERIVATIVES OF LIFT COEFFICIENT WITH
          RESPECT TO PITCH RATE.

F44     - THE TABLE OF DERIVATIVES OF SIDE FORCE COEFFICIENT WITH
          RESPECT TO YAW RATE.

F46     - THE TABLE OF DERIVATIVES OF LIFT COEFFICIENT WITH
          RESPECT TO ANGLE OF ATTACK RATE.

FSX     - X COMPONENT OF FORCE IN STABILITY AXES (LBS.).

FSY     - Y COMPONENT OF FORCE IN STABILITY AXES (LBS.).

FT1     - TABLE OF THRUST FORCE UNDER IDLE POWER (LBS.).
```

```
FT2     - TABLE OF THRUST FORCE UNDER MILITARY POWER (LBS.).

FT3     - TABLE OF THRUST FORCE UNDER MAXIMUM POWER (LBS.).

FWX     - THE X COMPONENT OF FORCE IN THE WIND AXES (LBS.).

FWY     - THE Y COMPONENT OF FORCE IN THE WIND AXES (LBS.).

FWZ     - THE Z COMPONENT OF FORCE IN THE WIND AXES (LBS.).

G       - ACCELERATION OF GRAVITY (FEET/SEC**2).

IA      - THE INDEX ON THE ANGLE OF ATTACK DIMENSION FOR THE
          AERODYNAMIC TABLES.

ICELL   - THE NUMBER OF THE SITUATION CELL IN WHICH THE TARGET
          PRESENTLY RESIDES.

IDT     - THE AMOUNT THE TIME INDEX IS INCREMENTED EACH TIME
          THROUGH THE INTEGRATION ROUTINE.

IM      - THE INDEX ON MACH NUMBER FOR THE AERODYNAMIC TABLES.

IRANDM  - THE INTEGER RANDOM NUMBER KERNEL FOR THE RANDU
          SUBROUTINE.

IT      - THE TIME INDEX FOR TIME SPENT IN A SINGLE MANEUVER,
          1 REPRESENTS .1 SEC.

ITIME   - THE OVERALL TIME INDEX, 1 REPRESENTS .1 SEC.

KBR     - A VARIABLE USED AS A SWITCH TO GET THE BARREL ROLL
          SEQUENCE STARTED.

KCELL   - THE NEW VALUE OF THE SITUATION CELL NUMBER WHICH IS
```

```
*              COMPARED TO ICELL TO TEST FOR CHANGE.                    *
*                                                                       *
*     KMN    - THE INDEX FOR THE BARREL ROLL ATTACK STAGES.             *
*                                                                       *
*     LEVEL  - REAL VARIABLE INDICATING THE SKILL LEVEL OF THE TARGET.  *
*                                                                       *
*     MANUV2 - THE NUMBER OF THE MANEUVER SELECTED WHEN TESTING TO SEE  *
*              IF THE SITUATION CELL HAS CHANGED.                       *
*                                                                       *
*     MANUVR - THE NUMBER OF THE MANUVR THE TARGET IS EXECUTING.        *
*                                                                       *
*     PB     - INITIAL VALUE OF ROLL RATE FOR THE TARGET (DEGREES/SEC). *
*                                                                       *
*     PBR    - ROLL RATE OF TARGET (RADIANS/SEC).                       *
*                                                                       *
*     PHID   - THE DESIRED VALUE OF BANK ANGLE (RADIANS).               *
*                                                                       *
*     PHID2  - THE MAXIMUM ANGLE OFF FROM WHICH A REAR ATTACK WILL BE   *
*              PRESSED, USED IN ISTATE (DEGREES).                       *
*                                                                       *
*     PHIDOT - THE RATE OF CHANGE OF BANK ANGLE (RADIANS/SEC).          *
*                                                                       *
*     PHIO   - BANK ANGLE IN DEGREES FOR OUTPUT.                        *
*                                                                       *
*     PHIROL - A PARAMETER USED IN DESIRE TO GENERATE THE DESIRED BANK  *
*              ANGLE.                                                   *
*                                                                       *
*     PHISTP - A PARAMETER USED IN DESIRE TO GENERATE THE DESIRED BANK  *
*              ANGLE.                                                   *
*                                                                       *
*     PHITE  - TARGET BANK ANGLE (DEGREES).                             *
*                                                                       *
*     PHITER - TARGET BANK ANGLE (RADIANS).                             *
*                                                                       *
*     PHITW  - TARGET BANK ANGLE IN THE WIND AXES (RADIANS).            *
```

```
*                                                                    *
*     PI     - 3.14159.                                              *
*                                                                    *
*     PSIF   - THE HEADING ANGLE OF THE OPPONENT EXPRESSED IN THE BODY *
*              AXIS SYSTEM OF THE TARGET (RADIANS).                  *
*                                                                    *
*     PSISTP - A PARAMETER USED IN DESIRE.                           *
*                                                                    *
*     PSITE  - THE TARGET HEADING (DEGREES).                         *
*                                                                    *
*     PSITER - THE TARGET HEADING (RADIANS).                         *
*                                                                    *
*     PSITW  - THE TARGET HEADING IN THE WIND AXES (RADIANS).        *
*                                                                    *
*     PW     - ROLL RATE IN THE WIND AXES (RADIANS/SEC).             *
*                                                                    *
*     QB     - THE INITIAL VALUE OF THE TARGET PITCH RATE (DEGREES/SEC)*
*                                                                    *
*     QBR    - TARGET PITCH RATE (RADIANS/SEC).                      *
*                                                                    *
*     QW     - PITCH RATE IN THE WIND AXES (RADIANS/SEC).            *
*                                                                    *
*     R      - RANGE BETWEEN THE TARGET AND THE OPPONENT (FEET).     *
*                                                                    *
*     RA     - THE FRACTIONAL STEP BETWEEN TWO ENTRIES IN THE        *
*              AERODYNAMIC TABLES, ALTITUDE OR ANGLE OF ATTACK, USED *
*              FOR INTERPOLATION.                                    *
*                                                                    *
*     RB     - INITIAL VALUE OF YAW RATE FOR THE TARGET (DEGREES/SEC). *
*                                                                    *
*     RBR    - TARGET YAW RATE (RADIANS/SEC).                        *
*                                                                    *
*     RDOT   - RATE OF CHANGE OF RANGE (FEET/SEC).                   *
*                                                                    *
*     RM     - THE FRACTIONAL STEP BETWEEN TWO ENTRIES IN THE        *
```

```
*            AERODYNAMIC TABLES, MACH NUMBER, USED FOR INTERPOLATION.*
*                                                                    *
*  RMAXT  - MAXIMUM TRACKING RANGE OF OPPONENT, USED IN ISTATE       *
*           (FEET).                                                  *
*                                                                    *
*  RMAXW  - MAXIMUM WEAPONS RANGE OF OPPONENT, USED IN ISTATE (FEET)*
*                                                                    *
*  RW     - TARGET YAW RATE IN THE WIND AXES (RADIANS/SEC).          *
*                                                                    *
*  SA     - SIN(A), USED IN CONVERSION OF AERODYNAMIC TABLES.        *
*                                                                    *
*  SCA    - SIN(A)*COS(A), USED IN THE CONVERSION OF THE AERODYNAMIC*
*           TABLES.                                                  *
*                                                                    *
*  SIDFRC - SIDE FORCE ON THE TARGET AIRCRAFT (LBS.).                *
*                                                                    *
*  SPH    - SIN(PHITER), USED IN DIRECTION COSINES FOR TRANSFOR-     *
*           MATION INTO BODY AXES.                                   *
*                                                                    *
*  SPS    - SIN(PSITER), USED IN DIRECTION COSINES FOR TRANSFOR-     *
*           MATION INTO BODY AXES.                                   *
*                                                                    *
*  SREF   - WING AREA (FEET**2).                                     *
*                                                                    *
*  SSA    - SINE SQUARED (A), USED IN CONVERSION OF AERODYNAMIC      *
*           TABLES.                                                  *
*                                                                    *
*  STH    - SIN(THETER), USED IN DIRECTION COSINES FOR TRANSFOR-     *
*           MATION INTO BODY AXES.                                   *
*                                                                    *
*  T      - IT*.1, USED TO OUTPUT TIME INTO A SPECIFIC MANEUVER.     *
*                                                                    *
*  TABX   - TABLE OF VALUES OF MACH NUMBER.                          *
*                                                                    *
*  TABLEX - TABLE OF VALUES OF ANGLE OF ATTACK.                      *
```

```
TDOTMX - THE MAXIMUM RATE OF CHANGE OF THRUST (LBS./SEC).

THED4  - THE MAXIMUM DEVIATION ANGLE ALLOWED FOR FRONT QUARTER
         ATTACK (DEGREES).

THED5  - THE MAXIMUM DEVIATION ANGLE ALLOWED FOR FRONT OFFENSE
         (DEGREES).

THEFER - PITCH ANGLE OF OPPONENT (RADIANS).

THETE  - PITCH ANGLE OF TARGET (DEGREES).

THETER - PITCH ANGLE OF TARGET (RADIANS).

THETW  - PITCH ANGLE OF TARGET IN WIND AXES (RADIANS).

THRSID - IDLE THRUST FOR TARGET (LBS.).

THRSML - MILITARY THRUST FOR TARGET (LBS.).

THRSMX - MAXIMUM THRUST FOR TARGET (LBS.).

THRSTD - DESIRED VALUE OF THRUST FROM SUBROUTINE DESIRE (LBS.).

THRUST - ACTUAL THRUST OF TARGET (LBS.).

TIMCHK - INDICATES AFTER WHAT INTERVAL OF TIME MANEUVER SHOULD BE
         REEVALUATED.

TIME   - TOTAL ELAPSED TIME, ITIME*.1.

TIMPUL - A PARAMETER IN DESIRE TO INDICATE THE VERTICAL ROLLING
         SCISSORS HAS BEEN INITIATED.
```

```
*      TMSTRT - SET IN DESIRE TO INDICATE THE AMOUNT OF DELAY BETWEEN     *
*               THE SELECTION OF A MANEUVER AND ITS INITIATION.            *
*                                                                          *
*      TMTERM - SET IN DESIRE TO INDICATE WHEN A MANEUVER SHOULD BE        *
*               TERMINATED.                                                *
*                                                                          *
*      TMTRCK - USED TO TALLY THE AMOUNT OF TIME THE TARGET HAS TRACKED    *
*               THE OPPONENT FOR A MISSLE OR GUN KILL.                     *
*                                                                          *
*      TX     - THE X COMPONENT OF THRUST IN THE BODY AXES.                *
*                                                                          *
*      TZ     - THE Z COMPONENT OF THRUST IN THE BODY AXES.                *
*                                                                          *
*      TXS    - THE X COMPONENT OF THRUST IN THE STABILITY AXES.           *
*                                                                          *
*      TZS    - THE Z COMPONENT OF THRUST IN THE STABILITY AXES.           *
*                                                                          *
*      UF     - THE X COMPONENT OF THE RELATIVE VELOCITY BETWEEN           *
*               AIRCRAFT (FEET/SEC).                                       *
*                                                                          *
*      UOPP   - THE X COMPONENT OF THE OPPONENT VELOCITY IN THE EARTH      *
*               REFERENCE (FT/SEC).                                        *
*                                                                          *
*      UTE    - THE X COMPONENT OF THE TARGET VELOCITY IN THE EARTH        *
*               REFERENCE (FT/SEC).                                        *
*                                                                          *
*      VEASTG - EQUIVALENT AIR SPEED OF THE TARGET (FT/SEC).               *
*                                                                          *
*      VELCOM - THE RELATIVE VELOCITY IN THE HORIZONTAL PLANE BETWEEN      *
*               THE AIRCRAFT.                                              *
*                                                                          *
*      VF     - THE Y COMPONENT OF THE RELATIVE VELOCITY BETWEEN THE       *
*               AIRCRAFT.                                                  *
*                                                                          *
*      VOPP   - THE Y COMPONENT OF THE OPPONENT VELOCITY IN THE EARTH      *
```

REFERENCE.

VT - TOTAL VELOCITY OF TARGET (FT/SEC).

VTE - THE Y COMPONENT OF TARGET VELOCITY IN THE EARTH REFERENCE (FT/SEC).

VTOPP - TOTAL VELOCITY OF THE OPPONENT (FT/SEC).

WF - THE Z COMPONENT OF THE RELATIVE VELOCITY BETWEEN THE AIRCRAFT.

WONE - VARIABLE USED TO MAINTAIN FLIGHT PATH CONTINUITY BETWEEN MANEUVERS.

WOPP - THE Z COMPONENT OF THE OPPONENT'S VELOCITY IN THE EARTH REFERENCE (FT/SEC).

WT - WEIGHT OF THE TARGET AIRCRAFT (LBS.).

WTE - THE Z COMPONENT OF TARGET VELOCITY IN THE EARTH REFERENCE (FT/SEC).

X - A SIX ELEMENT ARRAY COMPOSED OF THE STATE VARIABLES OF THE DIFFERENTIAL EQUATIONS TO BE INTEGRATED.

XF - THE X COMPONENT OF THE RELATIVE RANGE BETWEEN THE AIRCRAFT (FT).

XOPP - THE X COMPONENT OF THE OPPONENT'S POSITION IN THE EARTH REFERENCE (FT).

XTE - THE X COMPONENT OF THE TARGET'S POSITION IN THE EARTH REFERENCE (FT).

```
*      YF     - THE Y COMPONENT OF THE RELATIVE RANGE BETWEEN THE       *
*               AIRCRAFT (FT).                                          *
*                                                                       *
*      YOPP   - THE Y COMPONENT OF THE OPPONENT'S POSITION IN THE EARTH *
*               REFERENCE (FT).                                         *
*                                                                       *
*      YTE    - THE Y COMPONENT OF THE TARGET'S POSITION IN THE EARTH   *
*               REFERENCE (FT).                                         *
*                                                                       *
*      ZETAFR - THE AZIMUTH ANGLE OF THE OPPONENT AS VIEWED FROM THE    *
*               TARGET (RADIANS).                                       *
*                                                                       *
*      ZETDFR - THE AZIMUTH ANGLE RATE OF THE OPPONENT AS VIEWED FROM   *
*               THE TARGET (RADIANS/SEC).                               *
*                                                                       *
*      ZF     - THE Z COMPONENT OF THE RELATIVE RANGE BETWEEN THE       *
*               AIRCRAFT.                                               *
*                                                                       *
*      ZOPP   - THE Z COMPONENT OF THE OPPONENT'S POSITION IN THE EARTH *
*               REFERENCE.                                              *
*                                                                       *
*      ZTE    - THE Z COMPONENT OF THE TARGET'S POSITION IN THE EARTH   *
*               REFERENCE.                                              *
*                                                                       *
*                                                                       *
*************************************************************************
      COMMON/BLOCKG/ALDTMX,BEDTMX,TDOTMX,ALPHDT,PHIDOT,BETADT
      COMMON/BLOCKH/AMASS,FWX,THETW,PHITW,PSITW,RW,QW,PW
      COMMON /SKLL/PHID2,THED4,THED5,EFSD6,EPSD9,RMAXT,RMAXW            00000800
      COMMON/STATE/VFASTG/RANDM/IRANDM/BLOCKL/ALPMAX
      COMMON /DESIR/PSISTP,PHISTP,TIMPUL,R,XF,YF,ZF,ETAFR,EPSD,         00001200
     1PHIROL,ZETDFR,THEFER,ZOPP,CFAC,KMN,DEVANG,TMTRCK,ZETAFR,
     2THRUST,THRSMX,THRSML,THRSID,BETMAX,PHITER,ALPHAX,IT,THETER,       00001310
     3KRR,AMACHT,BETA,PSITER,VT,ZTE,VTOPP,RDOT,YTE,YOPP
      INTEGER TMSTRT,TMTERM,TIMCHK                                      00001330
```

```
      REAL LEVEL                                                        00001325
      DIMENSION AMACH(6),ALPHA(5),F1(6,6),F2(6,6),F3(6,6),F12(6,6),F34(600001400
     1,6),F37(6,6),F38(6,6),F39(6,6),F40(6,6),F41(6,6),F42(6,6),F43(6,6)00001500
     2,F44(6,6),F45(6,6),F46(6,6),XA(2),TABLEX(6),X(6),TABX(6),FT       00001600
     31(6,6),FT2(6,6),FT3(6,6)                                          00001700
      DIMENSION XT(200),YT(200),ZT(200),XO(200),YO(200),ZO(200)
************************************************************************
*     READ IN AERODYNAMIC DATA TABLES                                  *
************************************************************************
      READ (5,13) F1,F2,F3,F12,F34,F37,F38,F39,F40,F41,F42,F43,
     1F44,F45,F46,FT1,FT2,FT3,TABX,TABLEX                               00001900
   13 FORMAT (6F9.4)                                                    00002000
*
*
      KOUNT=1
      PI=3.14159                                                        00002200
      PI2=6.28318
      PIHALF=1.57079
      IRANDM=251                                                        00002300
      G=32.174                                                          00006500
*
*
************************************************************************
*   THE FOLLOWING ROUTINE CONVERTS THE AERODYNAMIC DATA FROM           *
*   STABILITY COORDINATES TO BODY AXES FOR THE MOMENT EQS              *
************************************************************************
      DO 6 I=1,6                                                        00002700
      A=TABLEX(I)*.0174533                                              00002800
      SA=SIN(A)                                                         00002900
      SSA=SA*SA                                                         00003000
      CA=COS(A)                                                         00003100
      CCA=CA*CA                                                         00003200
      SCA=CA*SA                                                         00003300
      DO 6 J=1,6                                                        00003400
      F34(I,J)=F34(I,J)*CA                                              00003500
```

```
      A1=F37(I,J)                                                       00003600
      A2=F42(I,J)                                                       00003700
      A3=F38(I,J)                                                       00003800
      A4=F43(I,J)                                                       00003900
      ASC1=(A1-A4)*SCA
      ASC2=(A2+A3)*SCA
      F37(I,J)=A1*CCA-ASC2+A4*SSA
      F42(I,J)=ASC1+A2*CCA-A3*SSA
      F38(I,J)=ASC1+A3*CCA-A2*SSA
    6 F43(I,J)=A4*CCA+ASC2+A1*SSA
      ITIME=0                                                           00004400
      TIME=0.0                                                          00004410
   50 FORMAT(24H ENTER PILOT SKILL LEVEL,F5.2)                          00004700
*********************************************************************
*     READ IN THE PILOT'S SKILL LEVEL                               *
*********************************************************************
      READ (5,51)LEVEL
      WRITE ( 6,50)LEVEL
   51 FORMAT (F4.2)                                                     00004900
      ALPHAD=0.                                                         00005000
      BETAD=0.                                                          00005100
*********************************************************************
*   SPECIFY THE BOUNDARIES OF ANGLE OF ATTACK, SIDESLIP, AND THEIR  *
*     RATES OF CHANGE                                               *
*********************************************************************
      ALPMAX=30./57.29578                                               00005200
      ALPMIN=-5.0/57.29578                                              00005205
      BETMAX=10./57.29578                                               00005300
      ALDTMX=20./57.29578                                               00005400
      BEDTMX=-ALPMIN
      TDOTMX=4500.                                                      00005600
*********************************************************************
*  THIS ROUTINE SETS DECISION PARAMETERS BASED ON PILOT SKILL LEVEL *
*********************************************************************
      CALL SKILL(LEVEL,ALPMAX)                                          00005625
```

```
*****************************************************************************
*  SPECIFY AIRCRAFT PARAMETERS                                              *
*****************************************************************************
C
      SPEF= 530.0                                                       00005700
      B=38.41                                                           00005800
      WT=41487.5                                                        00005900
      CBAR=16.042                                                       00006000
      EYEXX=26602.                                                      00006100
      EYEYY=128210.                                                     00006200
      EYEZZ=146629.                                                     00006300
      EYEXZ=3302.                                                       00006400
      ALPHAT=5.25/57.29578
      AMASS=WT/G                                                        00006600
*****************************************************************************
*        READ IN INITIAL AIRCRAFT STATES FOR THE TARGET AIRCRAFT           *
*****************************************************************************
    1 READ (5,5) XTE,YTE,ZTE,UTE,VTE,WTE,PSITE,THETE,PHITE,RB,QB,PB
    5 FORMAT (6F8.1/F9.1,F9.3,4F9.1)                                    00006900
      VT=SQRT(UTE**2+VTE**2+WTE**2)                                     00007200
      IF (ZTE.GT.0.0) GO TO 1                                           00007300
      TMSTRT=0                                                          00007400
      TIMCHK=100                                                        00007500
      IDT=5
*****************************************************************************
*  INITIALIZE THE VALUE OF ANGLE OF ATTACK                                  *
*  AND OTHER CONTROL VARIABLES                                              *
*****************************************************************************
      ALPHAX=3.134/57.29578                                             00007700
      ALPHO=3.134
      ALPHDT=0.                                                         00007900
      BETADT=0.0                                                        00008000
      PHIDOT=0.0                                                        00008100
      BETA=0.                                                           00008200
      BETO=0.
```

```
      THRUST=9180.                                                      00003400
*************************************************************************
*  WONE IS USED TO MAINTAIN FLIGHT PATH ANGLE CONTINUITY BETWEEN        *
*  MANEUVERS                                                            *
*************************************************************************
      WONE=WTE                                                          00003500
      PSITER=PSITE/57.29578                                             00003600
      THETER=THETE/57.29578                                             00003700
      PHITER=PHITE/57.29578                                             00003800
      PHITW=PHITER
      TR1=SIN(THETER)*COS(ALPHAX)*COS(BETA)+SIN(PHITER)*COS(THETER)*COS
     1(ALPHAX)*SIN(BETA)-COS(PHITER)*COS(THETER)*SIN(ALPHAX)
      THETW=ASIN(TR1)
      TR2=COS(THETER)*SIN(PSITER)*COS(ALPHAX)*COS(BETA)-COS(ALPHAX)*SIN
     1(BETA)*(SIN(PHITER)*SIN(THETER)*SIN(PSITER)+COS(PHITER)*COS(PSITER
     2))-SIN(ALPHAX)*(COS(PHITER)*SIN(THETER)*SIN(PSITER)-SIN(PHITER)*
     3COS(PSITER))
      PSITW=ASIN(TR2/COS(THETW))
      RPR=RP/57.29578                                                   00003900
      QRR=QB/57.29578                                                   00009000
      PRR=PB/57.29578                                                   00009100
*************************************************************************
*   FOR THE BARREL ROLL ATTACK SAVE KBR                                 *
*************************************************************************
      KBR=49                                                            00009300
   85 FORMAT(5X,4HTIME,5X,7HVT(FPS),9X,3HXTE,11X,3HYTE,11X,3HZTE,10X,5HA
     1LPHA,6X,4HBETA,8X,4HBANK,8X,5HPITCH,5X,7HHEADING)
   86 FORMAT(5X,F5.1,5X,F8.2,5X,F9.2,5X,F9.2,5X,F9.2,5X,F6.2,5X,F6.2,5X,
     1F7.2,5X,F7.2,5X,F7.2)
      PRINT 85
      WRITE(6,86)TIME,VT,XTE,YTE,ZTE,ALPHO,BETO,PHITE,THETE,PSITE
*************************************************************************
*   READ IN THE OPPONENT'S INITIAL STATES                               *
*************************************************************************
      READ (5,201) XOPP,YOPP,ZOPP,UOPP,VOPP,WOPP,PHIFER,THEFER,PSIFER
```

```
  201 FORMAT (9F9.1)
      XT(1)=XTE
      YT(1)=YTE
      ZT(1)=ZTE
      XO(1)=XOPP
      YO(1)=YOPP
      ZO(1)=ZOPP
      VTOPP=SQRT(UOPP*UOPP+VOPP*VOPP+WOPP*WOPP)                     00010600
      DELX=XOPP-XTE                                                  00011000
      DELY=YOPP-YTE                                                  00011100
      DELZ=ZOPP-ZTE                                                  00011200
      R=SQRT(DELX*DELX+DELY*DELY+DELZ*DELZ)                          00011300
      ANGOFF=ACOS((DELX*UOPP+DELY*VOPP+DELZ*WOPP)/(VTOPP*R))*57.29578
      DEVOPP=180.-ANGOFF
*********************************************************************
*  GET TERMS FOR TRANSFORMATION INTO BODY AXES OF THE TARGET       *
*********************************************************************
      CTH=COS(THETER)                                                00011800
      STH=SIN(THETER)                                                00011900
      CPS=COS(PSITER)                                                00012000
      SPS=SIN(PSITER)                                                00012100
      CPH=COS(PHITER)                                                00012200
      SPH=SIN(PHITER)                                                00012300
      C11=CTH*CPS                                                    00012400
      C12=CTH*SPS                                                    00012500
      C13=-STH                                                       00012600
      SPHSTH=SPH*STH
      CPHSTH=CPH*STH
      C21=SPHSTH*CPS-CPH*SPS
      C22=SPHSTH*SPS+CPH*CPS
      C23=SPH*CTH                                                    00012900
      C31=CPHSTH*CPS+SPH*SPS
      C32=CPHSTH*SPS-SPH*CPS
      C33=CPH*CTH                                                    00013200
*********************************************************************
```

```
*   PERFORM TRANSFORMATION INTO TARGET BODY AXES OF THE                  *
*   RELATIVE RANGE AND RANGE RATE VECTORS                                *
**************************************************************************
      XF=C11*DELX+C12*DELY +C13*DELZ                                       00013500
      YF=C21*DELX+C22*DELY+C23*DELZ                                        00013600
      ZF=C31*DELX+C32*DELY+C33*DELZ                                        00013700
      DELU=UOPP-UTE                                                        00013800
      DELV=VOPP-VTF                                                        00013900
      DELW=WOPP-WTE                                                        00014000
      UF=(C11*DELU+C12*DELV+C13*DELW)-(QBR*ZF-RBR*YF)
      VF=(C21*DELU+C22*DELV+C23*DELW)-(RBR*XF-PBR*ZF)
      WF=(C31*DELU+C32*DELV+C33*DELW)-(PBR*YF-QBR*XF)
**************************************************************************
*  THE AZIMUTH AND ELEVATION ANGLES ARE NOW DETERMINED                   *
*   FOR THE OPPONENT AS SEEN FROM THE TARGET                             *
**************************************************************************
      ETAFR=ASIN(-ZF/R)
      ZETAFR=ASIN(YF/(R*COS(ETAFR)))
      IF(XF.GE.0.)GO TO 2
      IF(ZETAFR.GE.0.)GO TO 3
      ZETAFR=-PI-ZETAFR
      GO TO 2
    3 ZETAFR=PI-ZETAFR
    2 DEVANG=ACOS(COS(ZETAFR)*COS(ETAFR))*57.29578
      VELCOM=SQRT(UF*UF+VF*VF)                                             00014800
      IF(VF.EQ.0.0) GO TO 41                                               00014810
      PSIF=ACOS(UF/VELCOM)
      IF(VF.LT.0.0) PSIF=-PSIF                                             00015000
      GO TO 42                                                             00015010
   41 PSIF=0.0                                                             00015020
   42 CETAFR=COS(ETAFR)
      PSIZET=ZETAFR-PSIF
      ZETDFR=VELCOM*COS(PIHALF+PSIZET)/(R*CETAFR)
      ETADF=(-VELCOM*COS(-PSIZET)*SIN(ETAFR)-WF*CETAFR)/R
      RDOT=UF*COS(ETAFR)*COS(ZETAFR)+VF*COS(ETAFR)*SIN(ZETAFR)-WF*SIN(ET
```

```
     1AFR)
      CALL ATMOS (ZTE,RHO,CA,SIGMA)
      VEASTG=VT*SQRT(SIGMA)
***********************************************************************
*   THE SITUATION CELL AND APPROPRIATE MANEUVER ARE NOW DETERMINED    *
***********************************************************************
      ICELL=ISTATE(R,DEVANG,ANGOFF,DEVOPP,ROOT,MANUVR)                  00015700
***********************************************************************
*    THIS IS THE TIME INTO A SINGLE MANUEVER                          *
***********************************************************************
 61   IT=0                                                              00015900
***********************************************************************
*                                                                     *
*  THIS IS THE BEGINNING OF THE MAIN LOOP OF THE PROGRAM              *
*                                                                     *
***********************************************************************
*
***********************************************************************
*  FIRST THE ATMOSPHERIC PARAMETERS ARE DETERMINED USING ALTITUDE     *
***********************************************************************
   14 CALL ATMOS (ZTE,RHO,CA,SIGMA)                                     00016100
      VT=SQRT((UTE*UTE)+(VTE*VTE)+(WTE*WTE))                            00016200
      AMACHT=VT/CA                                                      00016300
      DYNPRS=.5*RHO*(VT*VT)                                             00016400
      VEASTG=VT*SQRT(SIGMA)                                             00016500
      IF(ZTE.GT.-4000.)MANUVR=1
***********************************************************************
*  THE INDEX FOR THE MACH NUMBER IS DETERMINED FOR THE                *
*  INTERPOLATION OF THE AERODYNAMIC COEFFICIENTS                      *
***********************************************************************
   10 IF (AMACHT-2.2)100,101,101                                        00016700
  101 IM=6                                                              00016800
      GO TO 102                                                         00016900
  100 IF (AMACHT.GT.0.2)GO TO 103                                       00017000
      IM=1                                                              00017100
```

```
      GO TO 102                                                         00017200
  103 DO 104 I=1,6                                                      00017300
      IF (AMACHT-TABX(I))105,107,104                                    00017400
  104 CONTINUE                                                          00017500
  107 IM=I                                                              00017600
  102 RM=0.0                                                            00017700
      GO TO 106                                                         00017800
  105 IM=I-1                                                            00017900
      RM=(AMACHT-TABX(IM))/(TABX(IM+1)-TABX(IM))                        00018000
*************************************************************************
*  THE INDEX FOR THE ALTITUDE IS DETERMINED FOR THE INTERPOLATION       *
*   OF THE THRUST :  MAXIMUM THRUST  MILITARY THRUST  IDLE THRUST       *
*************************************************************************
  106 IA=ABS(ZTE/10000.)+1.0001                                         00018100
      IF (IA.GT.6) RA=0.0                                               00018200
      IF (IA.GT.6) IA=6                                                 00018300
      ZA=IA                                                             00018400
      IF(IA.LT.6)RA=(-ZTE)/10000.-ZA+1.                                 00018500
      THRSID=2000.*EXTRA(IA,IM,RM,RA,FT1)                               00018600
      THRSML=2000.*EXTRA(IA,IM,RM,RA,FT2)                               00018700
      THRSMX=2000.*EXTRA(IA,IM,RM,RA,FT3)                               00018800
      IF(WTE.EQ.WONE) GO TO 32
      WTWO=WTE-WONE
      IF(IT.LT.TMSTRT.AND.WTE.NE.0.0)ALPHAX=ALPHAX+SIGN(WTWO/100.*ALDTMX
     1*FLOAT(IDT)*.1,WTWO)
      IF (ALPHAX.GT.ALPMAX)ALPHAX=ALPMAX                                00019500
*************************************************************************
*        DETERMINE DESIRED CONTROL INPUTS                               *
*************************************************************************
   32 CALL DESIRE(MANUVR,ALPHAD,BETAD,PHID,THRSTD,TMSTRT,TIMCHK,TMTERM) 00019700
      ALPHO = 57.2957795*(ALPHAD)                                       00019800
      BETO = 57.2957795*(BETAD)                                         00019900
      PHIO = 57.2957795*(PHID)                                          00020000
   22 IF(IT.LT.TMSTRT) GO TO 17                                         00020300
*************************************************************************
```

```
*        DETERMINE ACTUAL CONTROL INPUTS                                    *
*****************************************************************************
      CALL CONTRL(ALPHAD,BETAD,PHID,THRSTD,IDT,ALPHAX,BETA,PHITER,          00020500
     1THRUST)                                                               00020600
*****************************************************************************
*   THE CONTROLS MUST BE KEPT WITHIN LIMITS                                 *
*****************************************************************************
      IF(ALPHAX.GT.ALPMAX)ALPHAX=ALPMAX                                     00020615
      IF(ALPHAX.LT.ALPMIN) ALPHAX=ALPMIN                                    00020620
      IF(ABS(BETA).GT.BETMAX) BETA=SIGN(BETMAX,BETA)                        00020625
      IF(THRUST.GT.THRSMX) THRUST=THRSMX                                    00020630
      IF(THRUST.LT.THRSID) THRUST=THRSID                                    00020635
      ALPHO = 57.2957795*(ALPHAX)                                           00020700
      BETO = 57.2957795*(BETA)                                              00020800
      PHIO = 57.2957795*(PHITER)                                            00020900
   17 TX=THRUST*COS(ALPHAT)                                                 00021200
      TZ=-THRUST*SIN(ALPHAT)                                                00021300
   25 CALL TRNSSB(TX,TZ,ALPHAX,TXS,TZS)                                     00021400
      ALPHO=57.295*ALPHAX
      IF (ALPHAX-ALPMAX)120,121,121                                         00021800
  121 IA=6                                                                  00021900
      GO TO 122                                                             00022000
  120 IF(ALPHAX.GT.ALPMIN) GO TO 123
      ALPHAX=TABLEX(1)/57.295779                                            00022200
      IA=1                                                                  00022300
      GO TO 122                                                             00022400
  123 DO 124 I=1,6                                                          00022500
      IF (ALPHAX*57.295779-TABLEX(I))126,125,124
  124 CONTINUE                                                              00022700
  125 IA=I                                                                  00022800
  122 PA=0.0                                                                00022900
      GO TO 127                                                             00023000
  126 IA=I-1                                                                00023100
      PA=(ALPHAX*57.2958-TABLEX(IA))/(TABLEX(IA+1)-TABLEX(IA))
*****************************************************************************
```

```
*  THE AERODYNAMIC COEFFICIENTS AND STABILITY DERIVATIVES ARE FOUND     *
**************************************************************************
 127  COEF2=EXTRA(IA,IM,RM,PA,F2)                                          00023400
      COEF3=EXTRA(IA,IM,RM,RA,F3)                                          00023500
      COEF12=EXTRA(IA,IM,PM,RA,F12)                                        00023600
      COEF39=EXTRA(IA,IM,PM,RA,F39)                                        00024000
      COEF41=EXTRA(IA,IM,RM,RA,F41)                                        00024200
      COEF44=EXTRA(IA,IM,PM,PA,F44)                                        00024500
      COEF46=EXTRA(IA,IM,PM,RA,F46)                                        00024700
      CL=COEF2+(COEF41*(QBR-ALPHDT)+COEF46*ALPHDT)*CBAR/(2.*VT)            00024800
      CLMAX=10.0/DYNPPS*WT/SREF                                            00024900
      IF(CL-CLMAX)23,23,24                                                 00025000
   24 ALPHAX = ALPHAX-(CL-CLMAX)/CLMAX*ALPHAX
      ALPHDT=-ALDTMX                                                       00025700
      GO TO 25                                                             00025800
   23 DYNSRF=DYNPPS*SREF
      ALIFT=CL*DYNSRF
      IF(THRSTD.GT.THRSID)COEF12=0.                                        00026000
      CD=COEF3+COEF12                                                      00026100
      DRAG=CD*DYNSRF
      CY=(COEF39*(PBR*COS(ALPHAX)+RBR*SIN(ALPHAX))+COEF44*(RBR*COS(ALP
     1HAX)-PBR*SIN(ALPHAX)))*B/(2.*VT)
      SIDFRC=CY*DYNSRF
      FSX=TXS-DRAG                                                         00027200
      FSY=-SIDFRC                                                          00027300
**************************************************************************
*  THE AERODYNAMIC FORCES IN THE WIND AXES ARE DETERMINED               *
**************************************************************************
      FWX=COS(BETA)*FSX+SIN(BETA)*FSY-WT*SIN(THETW)
      FWY=-SIN(BETA)*FSX+COS(BETA)*FSY+WT*SIN(PHITW)*COS(THETW)
      FWZ=TZS-ALIFT+WT*COS(THETW)*COS(PHITW)
**************************************************************************
*   EULER ANGLE RATES FOR THE WIND AXES                                 *
**************************************************************************
      AMVT=AMASS*VT
```

```
      RW=FWY/AMVT
      QW=-FWZ/AMVT
      PW=PBR*COS(ALPHAX)*COS(BETA)+(QBR-ALPHDT)*SIN(BETA)+RBR*SIN(ALPHAX
     1)*COS(BETA)
      X(1)=VT                                                           00023200
      X(2)=THETW                                                        00023300
      X(3)=PSITW                                                        00023400
      X(4)=XTE                                                          00023500
      X(5)=YTE                                                          00023600
      X(6)=ZTE                                                          00023700
*****************************************************************************
*         INTEGRATE THE EQUATIONS OF MOTION                                 *
*****************************************************************************
      CALL EULER(X,IDT,UTE,VTE,WTE,MANUVR,PHITER,THETER,KOUNT)
      ITIME=ITIME+IDT                                                   00029200
      IT=IT+IDT                                                         00029300
      VT=X(1)                                                           00029500
      THETW=X(2)                                                        00029600
      PSITW=X(3)
      PHITW=PHITER
      PRINT *,"THETW= ",THETW
      PRINT *,"PSITW= ",PSITW
      PRINT *,"PHITW= ",PHITW
      PRINT *,"ALPHAX= ",ALPHAX
      PRINT *,"BETA= ",BETA
      TR3=SIN(THETW)*COS(ALPHAX)*COS(BETA)-SIN(PHITW)*COS(THETW)*SIN
     1(BETA)-COS(PHITW)*COS(THETW)*SIN(ALPHAX)*COS(BETA)
      PRINT *,"TR3= ",TR3
      THETER=ASIN(TR3)
      TR4=COS(THETW)*SIN(PSITW)*COS(ALPHAX)*COS(BETA)+SIN(BETA)*(SIN(
     1PHITW)*SIN(THETW)*SIN(PSITW)+COS(PHITW)*COS(PSITW))+SIN(ALPHAX)*
     2COS(BETA)*(COS(PHITW)*SIN(THETW)*SIN(PSITW)-SIN(PHITW)*COS(PSITW))
      PRINT *,"TR4= ",TR4
      PSITER=ASIN(TR4/COS(THETER))
      PHITER=PHITW
```

```
      IF(THETER.GT.PIHALF)THETER=PI-THETER
      IF(THETER.LT.-PIHALF)THETER=-PI-THETER
      IF(PSITER.GT.PI2)PSITER=PSITER-PI2
      IF(PSITER.LT.0.)PSITER=PI2+PSITER
      IF(PHITER.LT.-PI)PHITER=PHITER+PI2
      IF (PHITER.GT.PI)PHITER=PHITER-PI2
      XTE=X(4)
      YTE=X(5)
      ZTE=X(6)
************************************************************************
*   EULER ANGLES FOR THE BODY AXES                                      *
************************************************************************
      PSITE=57.2957795*(PSITER)
      THETE=57.2957795*(THETER)
      PHITE=57.2957795*(PHITER)
      BETO=57.2957795*(BETA)
      ALPHO=57.2957795*(ALPHAX)
************************************************************************
*   OBTAIN ROTATIONAL RATES FOR THE BODY AXES                           *
************************************************************************
      AINT=PHIDOT-TAN(THETW)*(RW*COS(PHITER)+QW*SIN(PHITER))
      BINT=-QW*SIN(BETA)+AINT*COS(BETA)
      CINT=BETADT-PW
      QBB=ALPHDT+QW*COS(BETA)+AINT*SIN(BETA)
      PBB=CINT*SIN(ALPHAX)+BINT*COS(ALPHAX)
      RBB=-CINT*COS(ALPHAX)+BINT*SIN(ALPHAX)
      TIME=FLOAT(ITIME)*.1
      T =FLOAT(IT)*.1
************************************************************************
*                                                                       *
*  OUTPUT THE POSITION VELOCITY AND ORIENTATION                         *
*                                                                       *
************************************************************************
      WRITE(6,86)TIME,VT,XTE,YTE,ZTE,ALPHO,BETO,PHITE,THETE,PSITE
      WRITE(6,86)PW,QW,XF,YF,ZF,ALPHAD,RW,PHITW,THETW,PSITW
```

```
*************************************************************************
*   THE FOLLOWING ARRAYS ARE USED FOR PLOTTING PURPOSES                 *
*************************************************************************
      IF(ITIME.NE.5)GO TO 4
      II=2
      GO TO 9
    4 II=II+1
    9 XT(II)=XTE
      YT(II)=YTE
      ZT(II)=ZTE
      XO(II)=XOPP
      YO(II)=YOPP
      ZO(II)=ZOPP
*************************************************************************
*  REEVALUATE STATE TO DETERMINE SITUATION CELL                         *
*************************************************************************
      CALL OPPON(XOPP,YOPP,ZOPP,JOPP,VOPP,WOPP,PHIFER,THEFER,PSIFER,
     1VTOPP,KOUNT)
      VTOPP=SQRT(UOPP*UOPP+VOPP*VOPP+WOPP*WOPP)
      DELX=XOPP-XTE
      DELY=YOPP-YTE
      DELZ=ZOPP-ZTE
      R=SQRT(DELX*DELX+DELY*DELY+DELZ*DELZ)
      ANGOFF=ACOS((DELX*UOPP+DELY*VOPP+DELZ*WOPP)/(VTOPP*R))*57.29578
      PRINT *,"ANGOFF= ",ANGOFF
      DEVOPP=180.-ANGOFF
*************************************************************************
*  GET TERMS FOR TRANSFORMATION INTO BODY AXES OF THE TARGET            *
*************************************************************************
      CTH=COS(THETER)
      STH=SIN(THETER)
      CPS=COS(PSITER)
      SPS=SIN(PSITER)
      CPH=COS(PHITER)
      SPH=SIN(PHITER)
```

```
      C11=CTH*CPS                                                       00035500
      C12=CTH*SPS                                                       00035600
      C13=-STH                                                          00035700
      SPHSTH=SPH*STH
      CPHSTH=CPH*STH
      C21=SPHSTH*CPS-CPH*SPS
      C22=SPHSTH*SPS+CPH*CPS
      C23=SPH*CTH                                                       00036000
      C31=CPHSTH*CPS+SPH*SPS
      C32=CPHSTH*SPS-SPH*CPS
      C33=CPH*CTH                                                       00036300
************************************************************************
*   PERFORM TRANSFORMATION INTO TARGET BODY AXES OF THE RANGE AND      *
*   RANGE RATE                                                         *
************************************************************************
      XF=C11*DELX+C12*DELY+C13*DELZ                                     00036600
      YF=C21*DELX+C22*DELY+C23*DELZ                                     00036700
      ZF=C31*DELX+C32*DELY+C33*DELZ                                     00036800
      DELU=UOPP-UTE                                                     00036900
      DELV=VOPP-VTE                                                     00037000
      DELW=WOPP-WTE                                                     00037100
      UF=(C11*DELU+C12*DELV+C13*DELW)-(QBR*ZF-RBR*YF)
      VF=(C21*DELU+C22*DELV+C23*DELW)-(RBR*XF-PBR*ZF)
      WF=(C31*DELU+C32*DELV+C33*DELW)-(PBR*YF-QBR*XF)
************************************************************************
*   COMPUTE THE AZIMUTH AND ELEVATION OF THE OPPONENT IN TERMS         *
*   OF DEVIATIONS FROM THE X-Z PLANE AND X-Y PLANE OF THE TARGET       *
*   AIRCRAFT'S BODY AXES                                               *
************************************************************************
      ETAFR=ASIN(-ZF/R)
      ZETAFR=ASIN(YF/(R*COS(ETAFR)))
      IF(XF.GE.0.)GO TO 7
      IF(ZETAFR.GE.0.)GO TO 8
      ZETAFR=-PI-ZETAFR
      GO TO 7
```

```
8    ZETAFR=PI-ZETAFR
7    DEVANG=ACOS(COS(ZETAFR)*COS(ETAFR))*57.29578
     PRINT *,"DEVANG= ",DEVANG
     VELCOM=SQRT(UF*UF+VF*VF)                                        00038000
     IF(VF.EQ.0.0) GO TO 61                                          00038010
     PSIF=ACOS(UF/VELCOM)
     IF(VF.LT.0.0) PSIF=-PSIF                                        00038200
     PRINT *,"PSIF= ",PSIF
     GO TO 62                                                        00038210
61   PSIF=0.0                                                        00038220
62   ZETDFR=VELCOM*COS(PIHALF+ZETAFR-PSIF)/(R*COS(ETAFR))
     ETADF=(-VELCOM*COS(PSIF-ZETAFR)*SIN(ETAFR)-WF*COS(ETAFR))/R
     RDOT=UF*COS(ETAFR)*COS(ZETAFR)+VF*COS(ETAFR)*SIN(ZETAFR)-WF*SIN(ET
    1AFR)
*************************************************************************
*   PRINT OUT THE POSITION AND VELOCITY OF THE OPPONENT                 *
*************************************************************************
     WRITE(6,8F)TIME,VTOPP,XOPP,YOPP,ZOPP,PHIFER,THEFER,PSIFER
*************************************************************************
*   CHECK FOR CHANGE IN SITUATION CELL AND MANEUVER                     *
*************************************************************************
     IF(ITIME.GE.600)GO TO 87
     IF(IT.LT.TIMCHK)GO TO 14
     KCELL=ISTATE(R,DEVANG,ANGOFF,DEVOPP,RDOT,MANUV2)                00038600
     IF((IT.GE.TMTERM).OR.(KCELL.NE.ICELL)) GO TO 80                 00038700
     IF(ITIME.LE.600)GO TO 14
87   WRITE(3) MANUVR
     WRITE(3) II
     DO 11 K=1,IT
     WRITE(3) XT(K),YT(K),ZT(K)
11   WRITE(3) XO(K),YO(K),ZO(K)
     WRITE ( 6,60)
60   FORMAT (11H END OF RUN)                                         00039000
     STOP                                                            00039100
80   IF (MANUVP.EQ.MANUV2) GO TO 14                                  00039250
```

```
      MANUVR=MANUV2                                                     00039275
      ICELL=KCELL                                                       00039300
      WONE=WTE                                                          00039400
****KBR MUST BE SET TO 49 TO INSURE THAT A BARREL ROLL CAN BE PERFORMED*
      KBR=49
      GO TO 81                                                          00039600
      END
```

```
      SUBROUTINE DESIRE(MANUVR,ALPHAD,BETAD,PHID,THRSTD,TMSTRT,TIMCHK,  00044900
     1TMTERM)                                                           00045000
************************************************************************
*  THIS SUBROUTINE COMPUTES THE DESIRED CONTROL INPUTS TO FLY THE      *
*  SELECTED MANEUVER                                                   *
*                                                                      *
*     INPUTS:                                                          *
*           MANEUVER NUMBER                                            *
*                                                                      *
*     OUTPUTS:                                                         *
*           DESIRED ANGLE OF ATTACK                                    *
*           DESIRED SIDE SLIP ANGLE                                    *
*           DESIRED BANK ANGLE                                         *
*           DESIRED THRUST                                             *
*           TIMING CONDITIONS                                          *
*                                                                      *
************************************************************************
      INTEGER TMSTRT,TIMCHK,TMTERM                                      00045100
      COMMON/STATE/VEASTG/RAND1/IRANDM/BLOCKL/ALPMAX                    00045300
      COMMON/BLOCKH/AMASS,FWX,THETW,PHITW,PSITW,RW,QW,PW                00045400
      COMMON /DESIR/PSISTP,PHISTP,TIMPUL,R,XF,YF,ZF,ETAFR,EPSD,
     1PHIROL,ZETOFR,THEFER,ZOPP,CFAC,KMN,DEVANG,TMTRCK,ZETAFR,
     2THRUST,THRSMX,THRSML,THRSID,BETMAX,PHITER,ALPHAX,IT,THETER,       00045710
     3KRR,AMACHT,BETA,PSITER,VT,ZTE,VTOPP,ROOT,YTE,YOPP
      PI=3.1415927
      PIHALF=1.5707963
      CALL RANDU(IRANDM,YFL)                                            00045850
      IF(IT.EQ.0)CFAC=SIGN(1.,YFL-.5)                                   00045900
      GOTO(12,3,9,3,4,24,14,16,2,24,3,3,3,3,9,2,27,28,24,
     13,3,2,20,21,22,23,34,35,21,34,38,34,21,41,42,23,21,
     221,37,34,100),MANUVR                                              00046200
************************************************************************
*        THE FOLLOWING MANEUVER IS A HARD PULL UP                      *
************************************************************************
   12 ALPHAD=0.                                                         00046400
```

```
      PHID=0.                                                           00046500
      BETAD=0.                                                          00046600
      IF(ABS(PHITER-PHID).LE.1.)ALPHAD=ALPMAX                           00046700
      THRSTD=THRSID                                                     00046800
      TMSTRT=10                                                         00046900
      TIMCHK=100
      IF(THETW.GE.1.047198)TIMCHK=IT+5
      TMTERM=100                                                        00047200
      RETURN                                                            00047300
************************************************************************
*     THE FOLLOWING MANEUVER IS A SPLIT S                              *
************************************************************************
    9 TMSTRT=5                                                          00048500
      TIMCHK=150                                                        00048600
      TMTERM =200                                                       00048700
      IF(IT.EQ.0) PSISTP=PSITER                                         00048800
      IF(IT.EQ.0)PHISTP=PHITER                                          00048900
      PHID=SIGN(PI,PHISTP)                                              00049100
      BETAD=0.                                                          00049200
      IF(ABS(PHITER-PHID).LE..1) GO TO 10
      ALPHAD=ALPHAX                                                     00049400
      GO TO 11                                                          00049500
   10 ALPHAD=ALPMAX                                                     00049600
   11 THRSTD=THRSMX                                                     00049700
      IF(ABS(PSITER-PSISTP).GT.2.97)PHID=0.
      IF(THETER.LT.0.2)RETURN
      ALPHAD=0.0
      RETURN
************************************************************************
*     THE FOLLOWING MANEUVER IS A HARD TURN                            *
************************************************************************
    3 ALPHAD=ALPMAX                                                     00050100
      BETAD=0.                                                          00050200
      THRSTD = THRSMX
      IF(ZF.EQ.0.0.AND.YF.EQ.0.0)GO TO 70
```

```
      IF(YTE.GT.YOPP)PHID=4.*PI/9.
      IF(YTE.LT.YOPP)PHID=-4.*PI/9.
      GO TO 71
70    PHID=CFAC*4.*PI/9.
71    TMSTRT=5
      TIMCHK=125
      TMTERM=200                                                   00051000
      IF(MANUVR.EQ.2)GO TO 13
      RETURN                                                       00051100
*********************************************************************
*    THE FOLLOWING MANEUVER IS A HARD TURN FOLLOWED BY A REVERSAL      *
*********************************************************************
13    IF(IT.LT.100)RETURN
      PHID=-PHID
      RETURN
*********************************************************************
*    THE FOLLOWING MANEUVER IS A VERTICAL DIVE FOLLOWED BY A          *
*    HARD PULL-UP                                                     *
*********************************************************************
4     TMSTRT=5                                                     00051400
      TIMCHK=150
      THRSTD=THRSMX                                                00051600
      IF(IT.GT.105)GO TO 5
      IF(THETER.LE.-1.39626) GO TO 6
      ALPHAD=ALPMAX                                                00051900
      PHID=SIGN(PI,PHITER)                                         00052000
      GO TO 33                                                     00052100
6     ALPHAD=0.                                                    00052200
      PHID=0.                                                      00052300
      GO TO 33                                                     00052400
5     ALPHAD=ALPMAX                                                00052500
      PHID=0.
      THRSTD=THRSID                                                00052700
33    BETAD=0.                                                     00052800
      TMTERM=300                                                   00052900
```

```
      RETURN
*************************************************************************
*     THE FOLLOWING MANEUVER IS A HIGH G ROLL OVER                      *
*************************************************************************
   14 TMSTRT=5
      TIMCHK=200
      IF(IT.EQ.0) PHISTP=PHITER
      ALPHAD=ALPMAX
      BETAD=SIGN(BETMAX,PHISTP)
      PHID=PHITER-SIGN(PI/9.,PHISTP)
      THRSTD=THRSID
      TMTERM=300
      IF(IT.LT.TMSTRT+30) GO TO 15
      IF(ABS(PHITER-PHISTP).LE..2)GO TO 90
   15 IF(TMTERM.EQ.IT) TIMCHK=TMTERM
      IF(TMTERM.LE.IT) PHID=0.
      RETURN
   90 PHID=0.
      BETAD=0.
      TMTERM=IT
      GO TO 15
*************************************************************************
*     THE FOLLOWING MANEUVER IS A HIGH G ROLL UNDERNEATH                *
*************************************************************************
   16 TMSTRT=5
      IF(IT.GT.0) GO TO 18
      PHISTP=PHITER
   18 ALPHAD=ALPMAX
      BETAD=-SIGN(BETMAX,PHISTP)
      PHID=PHITER+SIGN(PI/9.,PHISTP)
      THRSTD=THRSID
      TMTERM=300
      IF(IT.LT.TMSTRT+30) GO TO 19
      IF(ABS(PHITER-PHISTP).LE..2)GO TO 80
   19 TIMCHK=TMTERM
```

```
      IF(TMTERM.LE.IT) PHID=0.
      RETURN                                                            00056900
   80 PHID=0.
      BETAD=0.
      TMTERM=IT
      GO TO 19
******************************************************************
*     THE FOLLOWING MANEUVER IS A MAXIMUM ENERGY MANEUVER        *
******************************************************************
    2 ALPHAD=0.                                                         00057100
      IF(THETER.LT.-1.0472) ALPHAD=(-1.0472-THETER)/PI*3.*ALPMAX
      IF((ZTE.GT.-3000.).OR.(AMACHT.GE.1.2))ALPHAD=.7*ALPMAX
      BETAD=0.                                                          00057300
      PHID=0.                                                           00057400
      THRSTD=THRSMX                                                     00057500
      TMSTRT=10                                                         00057600
      TIMCHK=100
      TMTERM=100
      RETURN                                                            00057900
******************************************************************
*     THE FOLLOWING MANEUVER IS A MAXIMUM RATE TURN              *
******************************************************************
   24 IF(IT.EQ.0) PSISTP=PSITER                                         00058100
      IF(IT.EQ.0) PHISTP=PHITER                                         00058200
      IF(ABS(PHITER-2.3561945).LE..1) GO TO 25
      IF(ABS(PHITER-0.78539816).LE..1)GO TO 25
      ALPHAD=0.                                                         00058400
      GO TO 26                                                          00058500
   25 ALPHAD=ALPMAX                                                     00058600
   26 BETAD=0.                                                          00058700
      PHID=.75*SIGN(PI,PHISTP)
      IF(ABS(PSITER-PSISTP).GE.PIHALF)PHID=SIGN(0.78539816,PHISTP)
      THRSTD=THRSMX                                                     00059000
      TMSTRT=10                                                         00059100
      TIMCHK=125
```

```
      TMTERM=200                                                        00059300
      RETURN
********************************************************************************
*     THE FOLLOWING MANEUVER IS A VERTICAL ROLLING SCISSORS                    *
********************************************************************************
   27 TMSTRT=10                                                         00059600
      IF(IT.EQ.0) TIMPUL=0.                                             00059700
      TIMCHK=200
      ALPHAD=ALPMAX                                                     00059900
      THRSTD=THRSMX                                                     00060000
      IF(THETEP.LE.ALPMAX-PIHALF) TIMPUL=FLOAT(IT)*.1
      IF(TIMPUL.GT.0.) GO TO 29                                         00060200
      BETAD=0.                                                          00060300
      PHID=SIGN(PI,PHITER)                                              00060400
      GO TO 30                                                          00060500
   29 THRSTD=THRSID                                                     00060600
      BETAD=-CFAC*BETMAX                                                00060700
      PHID=0.
   30 TMTERM=300                                                        00060900
      RETURN                                                            00061000
********************************************************************************
*     THE FOLLOWING MANEUVER IS A SCISSORS.                                    *
********************************************************************************
   28 ALPHAD=ALPMAX                                                     00061200
      THRSTD=THRSMX                                                     00061300
      IF(IT.GT.50)GO TO 31
      IF(ZETAFP.GE.0.)GO TO 36
      BETAD=0.                                                          00061500
      PHID=-PIHALF
      GO TO 31
   36 BETAD=0.
      PHID=PIHALF
   31 IF(RDOT.GE.0.)GO TO 39
      GO TO 32
   39 BETAD=SIGN(BETMAX,PHITER)
```

```
      PHID=-SIGN(PIHALF,PHITER)
   32 TMSTRT=10                                                           00062000
      TIMCHK=50                                                           00062100
      TMTERM=300                                                          00062200
      RETURN                                                              00062300
*******************************************************************
*     THE FOLLOWING MANEUVER IS A PURE PURSUIT CURVE                   *
*******************************************************************
   21 EPSD=0.                                                             00064400
      GO TO 40                                                            00064500
*******************************************************************
*     THE FOLLOWING MANEUVER IS A LEAD PURSUIT CURVE                   *
*******************************************************************
    22 EPSD=-.34907
      GO TO 40                                                            00064800
*******************************************************************
*     THE FOLLOWING MANEUVER IS A LAG PURSUIT CURVE                    *
*******************************************************************
   20 EPSD=.34907
   40 TMSTRT=5
      TIMCHK=50
      TMTERM=150
      DEVANG=DEVANG/57.29578
      YZ2=YF*YF+ZF*ZF+.1
      ARCSIN = ASIN(YF/SQRT(YZ2))
      PHID=ARCSIN+PHITER
      IF(ZF.GT.0.0) PHID=-ARCSIN+SIGN(PI,YF)+PHITER
      IF(XF.GE.0.)ALPHAD=ABS(ARCSIN/PIHALF)*ALPMAX
      IF(ABS(ARCSIN).LT..03)ALPHAD=ALPHAX
      IF(XF.LT.0.)ALPHAD=ALPMAX
      IF (ALPHAD.GT.ALPMAX) ALPHAD=ALPMAX
      IF (ALPHAD.LT.-.087266) ALPHAD=-.087266
      BETAD=0.                                                            00063500
      THRSTD=THRUST-(VT-VTOPP-.084*RANGF)*THRSMX/1000.
      IF(THRSTD.GT.THRSMX) THRSTD=THRSMX                                  00063700
```

```
      IF(THRSTD.LT.THRSID) THRSTD=THRSID                                00063800
      IF(EPSD)60,61,62
   60 IF(ABS(EPSD+DEVANG).LT..08727)PHID=0.
      DEVANG=DEVANG*57.29578
      RETURN
   61 IF(ABS(DEVANG).LT..08727)PHID=0.
      DEVANG=DEVANG*57.29578
      RETURN
   62 IF(ABS(DEVANG-EPSD).LT..08727)PHID=0.
      DEVANG=DEVANG*57.29576
      RETURN                                                            00064200
*************************************************************************
*     THE FOLLOWING MANEUVER IS A HI SPEED YO-YO                        *
*************************************************************************
   23 PHIROL=PIHALF
      IF(IT.EQ.0) PHISTP=PHITER                                         00065100
      TMSTRT=5
      TIMCHK=50
      TMTERM=150
      ALPHAD=ALFMAX                                                     00065200
      PHID=PHISTP-SIGN(PHIROL,PHISTP)                                   00065300
      BETAD=-(PHID-PHITER)/PHIROL*BETMAX                                00065400
      THRSTD=THRSID                                                     00065500
      IF(RDOT.LT.-127.)RETURN
      IF(RDOT.LT.-84.) GO TO 21
      GO TO 22                                                          00065700
*************************************************************************
*     THE FOLLOWING MANEUVER IS A BARREL ROLL ATTACK                    *
*************************************************************************
   34 THRSTD=THRSMX                                                     00065900
      IF(KBR.EQ.49) GO TO 49                                            00066000
      GO TO KMN,(51,52,53,20)                                           00066100
   49 TMSTRT=5                                                          00066200
      TMTERM=200                                                        00066300
      IF(IT.EQ.0) PHISTP=PHITER                                         00066400
```

```
      IF(ABS(ZETDFR).LE..02) GO TO 51
      ALPHAD=ALPMAX*10.*ZETDFR
      IF(ALPHAD.GT.ALPMAX) ALPHAD=ALPMAX
      PHID=SIGN(PIHALF-THEFER,ZETDFR)
      BETAD=(PHITER-PHID)/9.424778*BETMAX
      TIMCHK=20
      RETURN
   51 ASSIGN 51 TO KMN
***THE PARAMETER MUST BE CHANGED TO ALLOW THE MANEUVER TO PROCEED*******
      KRR=0
      ALPHAD=0.
      BETAD=0.
      PHID=-SIGN(THEFER,ZETDFR)
      TIMCHK=20
      IF(AMACHT.GE.1.4) ASSIGN 52 TO KMN
      RETURN
   52 IF(ETAFR.NE.-.1) ALPHAD=(ETAFR+.1)/.1*ALPMAX+ALPHAX
      IF(ALPHAD.GT.ALPMAX) ALPHAD=ALPMAX
      IF(ZTE.LT.ZOPP) GO TO 20
      BETAD=0.
      TIMCHK=50
      IF(AMACHT.NE.1.) THRSTD=(1.-AMACHT)*THRUST+THRUST
      IF(THRSTD.GT.THRSMX) THRSTD=THRSMX
      IF(THRSTD.LT.THRSID) THRSTD=THRSID
      IF(RANGF.LE.8000.) ASSIGN 53 TO KMN
      RETURN
   53 YZ2=YF*YF+ZF*ZF+.1
      ARCSIN = ASIN(YF/SQRT(YZ2))
      PHID=PHITER+ARCSIN
      IF(ZF.GT.0.) PHID=PHITER+PI-ARCSIN
      IF(ZF.GT.0..AND.YF.LT.0.) PHID=PHITER-PI-ARCSIN
      IF(ZTE.LT.ZOPP) ALPHAD=ALPMAX
      BETAD=(PHITER-PHID)/9.424778*BETMAX
      THRSTD=THRUST*(VTOPP-VT+184.)/VT+THRUST
      IF(THRSTD.GT.THRSMX) THRSTD=THRSMX
```

```
      IF(THRSTD.LT.THRSID) THRSTD=THRSID                                00069700
      ASSIGN 20 TO KMN                                                  00069800
      RETURN                                                            00069900
***********************************************************************
*     THE FOLLOWING MANEUVER IS A LOW SPEED YO-YO                     *
***********************************************************************
   35 THRSTD=THRSMX                                                     00070100
      TMSTRT=5                                                          00070200
      TIMCHK=50
      TMTERM=100                                                        00070400
      IF(ABS(ZETAFR).LT.0.5.AND.DEVANG.LE.35.)GO TO 55
      ALPHAD=ALPHAX                                                     00070600
      PHID=SIGN(PIHALF-THEFER,ZETAFR)
      BETAD=(PHITER-PHID)/PI*3.*BETMAX
      RETURN                                                            00070900
   55 ALPHAD=0.                                                         00071500
      BETAD=0.                                                          00071600
      PHID=PHITER
      RETURN                                                            00071800
***********************************************************************
*     THE FOLLOWING MANEUVER IS A CONTINUATION TO IMPROVE MANEUVERING *
*     POSITION                                                        *
***********************************************************************
   38 ALPHAD=ALPHAX                                                     00072100
      BETAD=BETA                                                        00072200
      PHID=PHITER                                                       00072300
      THRSTD=THRUST                                                     00072400
      TMSTRT=1                                                          00072500
      TIMCHK=5                                                          00072600
      TMTERM=50                                                         00072700
      RETURN                                                            00072800
***********************************************************************
*     THE FOLLOWING MANEUVER IS A MISSILE ATTACK                      *
***********************************************************************
   41 IF(IT.EQ.0) TMTRCK=0.0                                            00072910
```

```
      IF(DEVANG.LE.10.) TMTRCK=TMTRCK+.5
      IF(TMTRCK .GE. 3.) WRITE(6,56)
 56   FORMAT (83H THE TARGET HAS TRACKED HIS OPPONENT FOR SUFFICIENT TIM00073200
     1E TO ACHIEVE A MISSILE KILL.)                                     00073300
      GO TO 21                                                          00073400
*************************************************************************
*     THE FOLLOWING MANEUVER IS A GUN ATTACK                             *
*************************************************************************
  42  IF(IT.EQ.0) TMTRCK=0.0                                            00073550
      IF(ABS(ZETAFR).LE..08727.AND.ABS(ETAFR).LE..34907)
     1TMTRCK=TMTRCK+.5
      IF(TMTRCK .GE. 3.) WRITE(6,57)
 57   FORMAT (79H0THE TARGET HAS TRACKED HIS OPPONENT FOR SUFFICIENT TIM00073900
     1E TO ACHIEVE A GUN KILL.)                                         00074000
      GO TO 22                                                          00074100
*************************************************************************
*     THE FOLLOWING MANEUVER IS A HEAD ON GUN ATTACK                     *
*************************************************************************
  37  EPSD=0.174533
      GO TO 40                                                          00074400
 100  WRITE (6,101)
 101  FORMAT (36H NO APPROPRIATE MANEUVER DETERMINED.  )
      END
```

```
      FUNCTION ISTATE(R,THETA,PHI,TOPP,RR,MANUVR)
      COMMON/STATF/VEASTG/RANDM/IRANDM/BLOCKL/ALPMAX
      COMMON /SKLL/PHID2,THED4,THED5,EPSD6,EPSD9,RMAXT,RMAXW
************************************************************************
*   THIS FUNCTION DETERMINES THE TARGET'S SITUATION CELL AND SELECTS   *
*   AN APPPOPRIATE MANEUVER FOR THE SITUATION                          *
*                                                                      *
*   INPUTS                                                             *
*     RANGE,DEVIATION ANGLE OF THE TARGET, ANGLE OFF OF THE TARGET     *
*     DEVIATION ANGLE OF THE OPPONENT, AND RANGE RATE                  *
*   OUTPUTS                                                            *
*     SITUATION CELL AND MANEUVER                                      *
*                                                                      *
************************************************************************
C     ENTRY "BEGINNING" BLOCK
C
      IF (PHI.LT. PHID2) GO TO 150
      IF (THETA .GT. THED4 ) GO TO 120
C
C     FRONT ATTACK
C
      IF (R .LT. 20000.) GO TO 100
      INDX = 533
      MANUVR = 40
      GO TO 900
 100  IF (R .LT.12000.) GO TO 110
      INDX = 532
      MANUVR = 38
      GO TO 900
 110  INDX = 531
      MANUVR = 39
      GO TO 900
C
 120  IF (THETA .GT. THED5) GO TO 300
C
```

```
C      FRONT OFFENSE
C
       IF (R.LE. 12000.) GO TO 130
       INDX = 423
       MANUVR = 30
       GO TO 900
 130   IF (THETA .LE. 50.) GO TO 140
       INDX = 422
       MANUVR = 29
       GO TO 900
 140   INDX = 421
       MANUVR = 31
       GO TO 900
C
C      ENTRY BLOCK A
C
 150   IF (THETA .GT. ALPMAX*57.29578) GO TO 200
C
C      REAR ATTACK
C
       IF (PHI  .LT. 60.) GO TO 190
       INDX = 520
 154   IF (R.LT. 4000.) GO TO 160
       INDX = INDX+4
       MANUVR = 25
       IF(INDX.EQ.504) MANUVR = 37
       GO TO 900
 160   IF (R.LT. 2000.) GO TO 170
       INDX = INDX + 3
       MANUVR = 25
       IF(INDX.EQ.503) MANUVR = 34
       GO TO 900
 170   IF (R.LT. 800.) GO TO 180
       INDX = INDX + 2
       MANUVR = 35
```

```
      GO TO 900
  180 INDX = INDX+1
      MANUVR = 35
      GO TO 900
C
  190 IF (PHI .LT. 45.) GO TO 194
      INDX = 510
      GO TO 154
  194 INDX = 500
      GO TO 154
C
C     ENTRY BLOCK B
C
  200 CONTINUE
C
C     REAR OFFENSE
C
      IF (PHI .LT. 60.) GO TO 230
      IF (THETA .GT. 90.) GO TO 220
      IF (R.LT. 12000.) GO TO 210
      INDX = 412
      MANUVR = 27
      GO TO 900
  210 INDX = 411
      IF(-RR.LT.R*.028) MANUVR = 23
      IF(-RR.GE.R*.028) MANUVR = 28
      GO TO 900
  220 INDX = 461
      MANUVR = 27
      GO TO 900
C
  230 IF (THETA .LT. 90.) GO TO 240
      INDX = 451
      MANUVR = 28
      GO TO 900
```

```
  240 IF (R.LT. 6000.) GO TO 250
      INDX = 403
      MANUVR = 28
      GO TO 900
  250 IF (R.LT. 3000.) GO TO 260
      INDX = 402
      IF(-RR.GE.R*.028+127.) MANUVR = 23
      IF(-RR.LT.R*.028+127.) MANUVR = 24
      IF(-RR.LT.R*.028+84.) MANUVR = 25
      IF(-RR.LT.R*.028+42.) MANUVR = 28
      GO TO 900
  260 INDX = 401
      IF(-RR.LE. 84.) MANUVR = 25
      IF(-RR.GT. 84.) MANUVR = 24
      IF(-RR.GT. 127.) MANUVR = 25
      GO TO 900
C
C     ENTRY BLOCK C
C
  300 CONTINUE
      IF (TOPP .GT. EPS06) GO TO 380
      IF (R .GT. RMAXT) GO TO 360
C
C     EVASION
C
      IF (R.GT. 3000.) GO TO 350
      IF (R.GT. 2000.) GO TO 330
      IF (RR .LT. -84.) GO TO 320
      INDX = 201
      MANUVR = ISELCT(1,2,3,4,4,1)
      GO TO 900
  320 INDX = 202
      IF (VEASTG.GT.507.) MANUVR = 7
      IF (VEASTG.LE.507.) MANUVR = 8
      GO TO 900
```

```
  330 INDX = 210
  332 IF (RP.GT. 0.) GO TO 340
      INDX = INDX + 1
      MANUVR = ISELCT(4,4,5,6,4,4)
      GO TO 900
  340 INDX = INDX + 2
      MANUVR = ISELCT(3,4,5,6,4,4)
      IF (INDX.EQ.222) MANUVR = 9
      GO TO 900
  350 INDX = 220
      GO TO 332
  360 IF (R.GT. RMAXW) GO TO 370
      INDX = 301
      MANUVR = ISELCT(19,20,19,22,21,20)
      GO TO 900
  370 INDX = 431
      MANUVR = 32
      GO TO 900
C
  380 IF (THETA -EPSD9 .GT. TOPP) GO TO 390
C
C     OFFENSE
C
      INDX = 441
      MANUVR = 33
      GO TO 900
C
C     DEFENSE
C
  390 IF (R.LT. 5000.) GO TO 392
      INDX = 317
      MANUVR = 16
      GO TO 900
  392 IF (R.LT. 3000.) GO TO 394
      INDX = 312
```

```
      MANUVR = ISELCT(11,11,12,12,14,14)                                00095900
      GO TO 900                                                         00096000
  394 INDX = 311                                                        00096100
      MANUVR = ISELCT(10,13,14,15,17,18)                                00096200
C                                                                       00096300
  900 CONTINUE                                                          00096400
      ISTATE = INDX                                                     00096500
      WRITE(6,400) MANUVR,INDX
  400 FORMAT(26H THE MANEUVER SELECTED IS ,I2,5X,18H SITUATION CELL IS,
     1I4)
      RETURN                                                            00096900
      END
```

```
      SUBROUTINE EULER(X,IDT,P4,P5,P6,MANUVR,PHITER,THETER,KOUNT)
*********************************************************************
*   THIS ROUTINE PERFORMS AN EULER INTEGRATION ON THE STATE VECTOR  *
*                                                                   *
*   INPUTS                                                          *
*      THE STATE VECTOR - VELOCITY, THETAW, PSIW, X, Y, AND Z       *
*      THE TIME INCREMENT                                           *
*      COMMON BLOCKH - MASS, FWX, WIND EULER ANGLES, AND ANGULAR RATES *
*                                                                   *
*   OUTPUTS                                                         *
*      NEW STATE VECTOR                                             *
*      VELOCITY COMPONENTS  P4 - VX ; P5 - VY ; P6 - VZ             *
*                                                                   *
*********************************************************************
      DIMENSION X(6)                                                    00077700
      COMMON /BLOCKH/AMASS,FWX,THETW,PHITW,PSITW,RW,QW,PW               00077800
      PI=3.1415927
      DT = FLOAT(IDT)*.1                                                   77900
      P1=FWX/AMASS                                                      00078000
      CPHTW=COS(PHITW)
      SPHTW=SIN(PHITW)
      IF(ABS(X(2))-1.57079.GT.-.05)  GO TO 1
      P3=(QW*SPHTW+RW*CPHTW)/COS(X(2))
      GO TO 2                                                           00078300
    1 P3=PW                                                             00078400
    2 P2=QW*CPHTW-RW*SPHTW
      X(1)=X(1)+P1*DT                                                   00078600
      X(2)=X(2)+P2*DT                                                   00078700
      X(3)=X(3)+P3*DT                                                   00078800
      IF(MANUVR.EQ.3.OR.MANUVR.EQ.15)GO TO 4
      GO TO 3
4     IF(THETER.GT.-1.48353)GO TO 3
      IF(KOUNT.EQ.2)GO TO 3
      X(2)=-1.92
      X(3)=X(3)+PI
```

```
      PHITER=PHITER-PI
      KOUNT=KOUNT+1
    3 CX2=COS(X(2))
      P4=X(1)*CX2*COS(X(3))
      P5=X(1)*CX2*SIN(X(3))
      P6=-X(1)*SIN(X(2))                                    00079100
      X(4)=X(4)+P4*DT                                       00079200
      X(5)=X(5)+P5*DT                                       00079300
      X(6)=X(6)+P6*DT                                       00079400
    5 RETURN
      END
```

```
      SUBROUTINE CONTRL(ALPHAD,BETAD,PHID,THRSTD,IDT,ALPHAX,BETA,PHITER,00074900
     1THRUST)                                                           00075000
      COMMON/BLOCKG/ALDTMX,BEDTMX,TDOTMX,ALPHDT,PHIDOT,BETADT           00075100
C                                                                       00075110
************************************************************************
*   THIS ROUTINE DETERMINES THE NEW VALUE OF THE CONTROL VARIABLES     *
*   BASED ON THE PRESENTLY DESIRED VALUES AND THE PRESENT ACTUAL VALUES*
*                                                                      *
*  INPUTS                                                              *
*    DESIRED VALUES OF THE FOUR CONTROLS                               *
*    ACTUAL VALUES OF THE FOUR CONTROLS                                *
*    THE INTEGRATION INCREMENT                                         *
*    THE MAXIMUM RATES FOR ALPHA, BETA, AND THRUST ARE IN COMMON       *
*                                                                      *
*    THE MAXIMUM RATE OF CHANGE OF PHI IS COMPUTED BECAUSE IT IS VERY  *
*    DEPENDENT ON ALPHA                                                *
*                                                                      *
*  OUTPUTS                                                             *
*    NEW ACTUAL VALUES FOR THE CONTROLS                                *
*    RATES FOR ALPHA, BETA, AND PHI                                    *
*                                                                      *
*   THE RELATIONSHIP BETWEEN THE DESIRED AND ACTUAL HAS A LINEAR       *
*   PORTION, A PARABOLIC PORTION, AND A CONSTANT PORTION               *
*                                                                      *
************************************************************************
C
      DT = FLOAT(IDT)*.1                                                   75295
      PHDTMX=3.14159-10.9*ALPHAX+10.8862*ALPHAX*ALPHAX                  00075300
      ERALPH=ALPHAD-ALPHAX
      IF(ABS(ERALPH).LE.1.0E-5) ERALPH=0.0                              00075410
      ERBETA=BETAD-BETA
      IF(ABS(ERBETA).LE.1.0E-5) ERBETA=0.0                              00075510
      ERRPHI=PHID-PHITER
      IF(ABS(ERRPHI).LE.1.0E-5) ERRPHI=0.0                              00075610
      ERTHRS=THRSTD-THRUST
```

```
IF(ABS(ERTHRS).LE.1.0E-6) ERTHRS=0.0                                 00075710
ALPHDT=ALDTMX*(ERALPH/ALDTMX*4.)**2
BETADT=BEDTMX*(ERBETA/BEDTMX*3.)**2
PHIDOT=PHDTMX*(ERPPHI/PHDTMX*12.)**2
THRSDT=TDOTMX*(ERTHRS/TDOTMX*3.)**2
IF(ALPHDT.LE.ABS(ERALPH))ALPHDT=ABS(ERALPH)
IF(ALPHDT.GE.ALDTMX)ALPHDT=ALDTMX                                    00076300
ALPHAX=ALPHAX+SIGN(ALPHDT*DT,ERALPH)                                 00076400
IF(BETADT.LE.ABS(ERBETA))BETADT=ABS(ERBETA)
IF(BETADT.GE.BEDTMX)BETADT=BEDTMX                                    00076600
BETA=BETA+SIGN(BETADT*DT,ERBETA)                                     00076700
IF(PHIDOT.LE.ABS(ERRPHI))PHIDOT=ABS(ERRPHI)
IF(PHIDOT.GE.PHDTMX)PHIDOT=PHDTMX                                    00076900
PHITER=PHITER+SIGN(PHIDOT*DT,ERRPHI)                                 00077000
IF(THRSDT.LE.ABS(ERTHRS))THRSDT=ABS(ERTHRS)
IF(THRSDT.GE.TDOTMX)THRSDT=TDOTMX                                    00077200
THRUST=THRUST+SIGN(THRSDT*DT,ERTHRS)                                 00077300
RETURN                                                               00077400
END
```

```
      SUBROUTINE OPPON(XOPP,YOPP,ZOPP,UOPP,VOPP,WOPP,PHIFER,THEFER,PSIFE
     1R,VTOPP,KOUNT)
*************************************************************************
*     THIS ROUTINE UPDATES THE OPPONENT'S STATES FROM THE FOLLOWING STATE*
*     EQUATIONS:                                                        *
*************************************************************************
C
      PI=3.14159
      PI2=6.28318
      IF(KOUNT.NE.1)GO TO 20
      IF(PSIFER.GT.PI2)GO TO 20
      UOPP=VTOPP*COS(PSIFER)
      VOPP=VTOPP*SIN(PSIFER)
      WOPP=0.
      XOPP=XOPP+UOPP*0.5
      YOPP=YOPP+VOPP*0.5
      ZOPP=ZOPP+WOPP*0.5
      PSIFER=PSIFER+PI/18*0.5
      PHIFER=PI/2.
      THEFER=0.0
      GO TO 30
   20 UOPP=900.
      VOPP=0.
      WOPP=0.
      XOPP=XOPP+UOPP*0.5
      YOPP=YOPP+VOPP*0.5
      ZOPP=ZOPP+WOPP*0.5
      PHIFER=0.
      PSIFER=0.
      THEFER=0.
      KOUNT=KOUNT+1
   30 RETURN
      END
```

```
      SUBROUTINE TRNSSB(XB,ZB,ALPHA,XS,ZS)
C
**********************************************************************
*    THIS SUBROUTINE TRANSFORMS A VECTOR IN THE BODY FIXED REFERENCE   *
*    FRAME INTO THE STABILITY AXIS SYSTEM                              *
**********************************************************************
C
C    INPUTS REQUIRED ARE:
C        XB AND ZB ARE THE VECTOR COMPONENTS IN THE BODY FIXED FRAME
C        ALPHA IS THE ANGLE OF ATTACK OF THE AIRCRAFT
C
C    OUTPUTS ARE:
C        XS AND ZS ARE THE VECTOR COMPONENTS IN THE STABILITY AXIS SYSTEM
C
      SNALPH=SIN(ALPHA)
      CSALPH=COS(ALPHA)
      XS=XB*CSALPH+ZB*SNALPH
      ZS=-XB*SNALPH+ZB*CSALPH
      RETURN
      END
```

```
      SUBROUTINE ATMOS(ZI,RHO,VS,SIGMA)
C
*******************************************************************
*   THE FOLLOWING SUBROUTINE COMPUTES ATMOSPHERIC PARAMETERS      *
*   OUTPUT VARIABLES:                                             *
*       RHO   -AIR DENSITY IN SLUGS/CUBIC FOOT                    *
*       VS     VELOCITY OF SOUND IN FEET/SECOND                   *
*       SIGMA-  DENSITY RATIO  (PRESENT ALTITUDE)/(SEA LEVEL)     *
*   INPUT VARIABLE IS ZI THE ALTITUDE IN FEET (ABOVE EARTH IS NEGATIVE)*
*                                                                 *
*   THE ATMOSPHERIC MODEL USED IS THE TWO LOWEST LAYERS OF THE    *
*   U.S. STANDARD ATMOSPHERE 1962                                 *
*                                                                 *
*******************************************************************
C
      REAL K1,K2,K3
      DIMENSION K1(2),HB(2),K2(2),K3(2),TB(2),RHOB(2)
      DATA K1(1),K1(2)/-.2255877E-4,0.0/,HB(1),HB(2)/0.0,10999.474/,K2(1
     1),K2(2)/-5.255871,0.0/,K3(1),K3(2)/0.0,.1576958E-3/,TB(1),TB(2)/
     A518.67,389.97/,RHOB(1),RHOB(2)/.2377E-2,.7061512E-3/
      IF (ZI.GT.0.0) GO TO 20
      TMP=.3048*(-ZI)
      HGP=TMP/(1.+TMP/6356766.)
      IALT=1
C    IALT INDICATES THE LAYER NUMBER
      IF(HGP.GE.11000.)IALT=2
      HDIFF=HGP-HB(IALT)
      TM=TB(IALT)*(1.0+K1(IALT)*HDIFF)
      RHO=RHOB(IALT)*(1.0+K1(IALT)*(HGP-HB(IALT)))**(-1.0-K2(IALT))
      IF (IALT.EQ.2) RHO=RHOB(2)*EXP(-K3(2)*HDIFF)
      VS=49.021175*SQRT(TM)
      SIGMA=RHO/.002378
      RETURN
   20 WRITE (6,40)
   40 FORMAT(26H  ALTITUDE IS SUBTERRANEAN)
```

```
      RETURN                                                    00043400
      END
```

```
      FUNCTION EXTRA(IA,JM,RM,RA,C)                                     00043600
C
************************************************************************
*  THIS ROUTINE INTERPOLATES THE VALUE OF A COEFFICIENT FROM           *
*  A TWO DIMENSIONAL TABLE                                              *
*                                                                       *
*  INPUTS                                                               *
*     THE SUBSCRIPTS OF THE GREATEST LOWER BOUND  IA & JM              *
*     THE FRACTIONS OF THE STEPS TO BE INTERPOLATED  RA & RM           *
*     THE TABLE TO BE USED IN THE INTERPOLATION                        *
*                                                                       *
*  OUTPUTS                                                              *
*     THE INTERPOLATED VALUE OF THE FUNCTION                           *
*                                                                       *
************************************************************************
C
      DIMENSION C(6,6)                                                  00043700
      IF (IA.GE.5.AND.(JM.GE.6))GO TO 10                                00043800
      IF (IA.GE.6) GO TO 9                                              00043900
      IF (JM.GE.6) GO TO 20                                             00044000
      DELC1=C(IA,JM+1)-C(IA,JM)
      DELC2=C(IA+1,JM+1)-C(IA+1,JM)                                     00044200
    9 DELC3=C(IA+1,JM)-C(IA,JM)                                         00044300
   10 EXTRA=C(IA,JM)+RA*(DELC3+RM*(DELC2- DELC1))+RM*DELC1             00044400
      RETURN                                                            00044500
   20 DELC1=C(IA+1,JM)-C(IA,JM)                                         00044600
      GO TO 10                                                          00044700
      END
```

```
      SUBROUTINE RANDU(IX,YFL)                                          00099900
*********************************************************************
*    THIS ROUTINE COMPUTES A UNIFORMLY DISTRIBUTED RANDOM INTEGER    *
*    AND A REAL NUMBER BETWEEN 0.0 AND 1.0                           *
*   INPUT                                                            *
*     AN INTEGER KERNEL IX                                           *
*                                                                    *
*   OUTPUTS                                                          *
*     A RANDOM INTEGER IX                                            *
*     A RANDOM REAL YFL                                              *
*                                                                    *
*   THE CDC 6600 ROUTINES RANF,RANSET, AND RANGET ARE UTILIZED TO    *
*    GENERATE THESE RANDOM NUMBERS                                   *
*                                                                    *
*********************************************************************
      CALL RANSET(IX)
      YFL=RANF(DUM)
      CALL RANGET(N)
      IX=N
      RETURN                                                            00100500
      END
```

```
      FUNCTION ISELCT(N1,N2,N3,N4,N5,N6)                                    00097100
C                                                                           00097150
*****************************************************************************
*   THIS FUNCTION SELECTS A MANEUVER RANDOMLY FROM 6 CHOICES               *
*****************************************************************************
C
      COMMON/STATE/VEASTG/RANDM/IRANDM/BLOCKL/ALPMAX                        00097300
      CALL RANDU(IRANDM,YFL)
      IF(YFL.GT..2) GO TO 10                                                00097500
      ISELCT=N1                                                             00097600
      RETURN                                                                00097700
   10 IF(YFL.GT..4) GO TO 20                                                00097800
      ISELCT=N2                                                             00097900
      RETURN                                                                00098000
   20 IF(YFL.GT..6) GO TO 30                                                00098100
      ISELCT=N3                                                             00098200
      RETURN                                                                00098300
   30 IF(YFL.GT..8) GO TO 40                                                00098400
      ISELCT=N4                                                             00098500
      RETURN                                                                00098600
   40 IF(YFL.GT..9) GO TO 50                                                00098700
      ISELCT=N5                                                             00098800
      RETURN                                                                00098900
   50 ISELCT=N6                                                             00099000
      RETURN                                                                00099100
      END
```

```
      SUBROUTINE SKILL(LEVEL,ALPMAX)
      COMMON /SKLL/PHID2,THED4,THED5,EPSD6,EPSD9,RMAXT,RMAXW
C
*****************************************************************
*    THIS SUBROUTINE ALLOWS FOR VARIATIONS IN JUDGMENT OF THE   *
*    TARGET PILOT.                                              *
*                                                               *
*    INPUT REQUIRED IS LEVEL, AN INTEGER SKILL FACTOR BETWEEN   *
*    -1 AND +1, ZERO REPRESENTS THE IDEAL PILOT.                *
*                                                               *
*    OUTPUTS ARE:                                               *
*      PHID2,ALPMAX,THED4,THED5,EPSD6,EPSD9,RMAXT, AND RMAXW.   *
*    THESE PARAMETERS ARE USED IN THE ISTATE SUBROUTINE TO      *
*    MAKE THE SITUATION CELL AND MANEUVER SELECTIONS.           *
*                                                               *
*****************************************************************
C
      REAL LEVEL
      IF(LEVEL.LT.-1.0) LEVEL=-1.0
      IF(LEVEL.GT. 1.0) LEVEL= 1.0
      PHID2=LEVEL*30.+90.
      IF(LEVEL.GT.0.0) ALPMAX=ALPMAX*1.33
      IF(LEVEL.LT.0.0) ALPMAX=ALPMAX*.5
      THED4=30.*LEVEL+30.
      THED5=30.*LEVEL+60.
      EPSD6=-10.*LEVEL+20.
      EPSD9=10.*LEVEL+10.
      RMAXT=-LEVEL*2000.+6000.
      RMAXW=-LEVEL*10000.+20000.
      RETURN
      END
```

Appendix B

Smart Target Aerodynamic Data

As discussed in Chapter VII, the moment equations are not used in the Smart Target model. Thus, the only aerodynamic coefficients and stability derivatives needed are those for the force equations.

The equations specifying these coefficients are as follows:

$$C_L = C_{L_b} + \frac{\bar{c}}{2V}\left(C_{L_q} q + C_{L_{\dot{\alpha}}} \dot{\alpha}\right) \tag{76}$$

$$C_D = C_{D_b} + \Delta C_{D_{SB}} \tag{77}$$

$$C_Y = \frac{b}{2V}\left(C_{Y_p} p + C_{Y_r} r\right) \tag{78}$$

where,

C_L = coefficient of lift

C_D = coefficient of drag

C_Y = coefficient of side force

$\bar{c}$ = mean aerodynamic chord

b = wing span

V = total velocity

These aerodynamic coefficients and stability derivatives are tabulated as functions of Mach number and angle of attack. Table IV lists the conventional designation, the definition, and a representative value from the tables at $M = .90$ and $\alpha = 5^{\circ}$.

Table IV
Aerodynamic Coefficients and Stability Derivatives

Designation	Definition	Representative Value
C_{L_b}	Basic lift force coefficient with elevator zero	.438
C_{D_b}	Basic drag coefficient with elevator zero	.052
$C_{D_{SB}}$	Drag coefficient increment due to full speed brake deflection	.050
C_{Y_p}	Derivative of side force coefficient with respect to roll rate	.168
C_{L_q}	Derivative of lift coefficient with respect to pitch rate	4.72
C_{Y_r}	Derivative of side force coefficient with respect to yaw rate	.72
$C_{L_{\dot{\alpha}}}$	Derivative of lift coefficient with respect to angle of attack rate	1.30

The aerodynamic forces along the stability axes are determined by the following:

$$x_S = - \bar{q} \, S \, C_D + T_{x_S} \tag{79}$$

$$y_S = - \bar{q} \, S \, C_Y \tag{80}$$

$$z_S = - \bar{q} \, S \, C_L + T_{z_S} \tag{81}$$

where,

$\bar{q}$ = dynamic pressure, $\frac{1}{2} \rho V^2$

S = reference area

T_{x_S} and T_{z_S} = thrust along the x and z stability axes, respectively.

These force equations written in the stability axes are resolved into the wind axes by the following transformation:

$$x_W = x_S \cos\beta + y_S \sin\beta - mg \sin\theta_W \tag{82}$$

$$y_W = -x_S \sin\beta + y_S \cos\beta + mg \sin\phi_W \cos\theta_W \tag{83}$$

$$z_W = z_S + mg \cos\theta_W \cos\phi_W \tag{84}$$

Eqs (82), (83), and (84) are then utilized to compute the Euler angle rates for the wind axes.

Appendix C

Smart Target Test Results

This appendix contains the results obtained in the simulation of each Air Combat Maneuver. A discussion of the desired controls to fly each maneuver is given, along with a tabular listing of the results obtained. Refer to Chapter IV for a detailed description of each maneuver and to Appendix A, SUBROUTINE DESIRE, for the exact controls used.

Evasive Maneuvers

Hard Pull Up. The hard pull up was programmed as a wings level climb at maximum angle of attack using idle thrust. The results of this maneuver are shown in Table V, based on pitch angle attained. As desired, angle of attack increased rapidly, with a corresponding increase in pitch, and a significant loss in airspeed.

Table V
Hard Pull Up

Pitch Angle - θ (deg)	Angle of Attack - α (deg)	Velocity - V_T (ft/sec)	Time (sec)
0.0	3.13	750.00	0.00
10.0	10.61	672.95	1.35
20.0	16.81	662.08	1.50
30.0	23.61	636.82	1.84
40.0	28.14	595.90	2.41
50.0	29.60	526.17	3.50
60.0	29.95	442.42	5.00
70.0	30.00	297.84	8.50

Split S. The split s was simulated by utilizing a maximum roll rate to achieve an inverted attitude. Maximum angle of attack was then applied to cause the target to turn in the vertical plane. The desired result was a heading change of 180° plus a slight increase in airspeed due to the addition of the gravity vector in the vertical turn. Table VI shows how the flight path changed based on pitch angle.

Table VI
Split S

Pitch Angle-θ (deg)	Bank Angle-ϕ (deg)	Heading Angle-ψ (deg)	Velocity-V_T (ft/sec)	Time (sec)
0.0	0.0	0.0	550.00	0.0
-30.0	179.5	0.0	558.94	4.5
-60.0	180.0	0.0	552.74	6.5
-90.0	0.0	180.0	573.65	8.5
-60.0	0.0	180.0	606.26	10.5
-30.0	0.0	180.0	632.43	12.5
0.0	0.0	180.0	638.17	15.0

Hard Turn. By commanding a maximum angle of attack turn with a bank angle of 80°, the hard turn simulated in Smart Target proved very effective. The results showed that a turn using these controls provided a heading change of 240° in 20 seconds.

Hard Turn Followed by a Turn Reversal. The controls used to simulate this maneuver were the same as the hard turn. The initial direction of turn was selected at random, and the turn reversal was accomplished after a 10 second delay. In a 20 second time interval the target was able to generate 130° heading change in one direction followed by an 88° heading change in the opposite direction.

High G Roll Over. To simulate the high g roll over, maximum angle of attack, bank angle change of 40^{o} per second, maximum sideslip angle, and idle thrust were used. The desired results were a vector roll of 360^{o}, with a large decrease in velocity. Table VII shows the results of the high g roll over.

Table VII
High G Roll Over

Bank Angle-ϕ (deg)	Pitch Angle-θ (deg)	Velocity-V_T (ft/sec)	Time (sec)
90.0	0.0	875.0	0.0
45.0	26.0	645.0	2.3
0.0	41.0	525.0	4.3
-45.0	35.0	450.0	5.8
-90.0	11.0	360.0	8.0
-135.0	-20.0	304.5	10.0
-180.0	-40.0	279.0	11.8
135.0	-46.0	275.0	13.5
90.0	-36.0	289.0	15.5

High G Roll Underneath. The high g roll underneath was accomplished using the same controls as the high g roll over. However, in a turning fight with low airspeed, the roll must be initiated in the direction of bank. In an attempt to turn over the top with this low airspeed, the aircraft would stall, and the maneuver would have to be aborted. The results of the high g roll underneath are shown in Table VIII.

Table VIII
High G Roll Underneath

Bank Angle-ϕ (deg)	Pitch Angle-θ (deg)	Velocity-V_T (ft/sec)	Time (sec)
90.0	0.0	675	0.0
135.0	-40.0	548	2.5
180.0	-62.0	485	4.2
-135.0	-69.0	456	6.0
-90.0	-58.0	442	8.0
-45.0	-38.0	434	10.0
0.0	-27.0	426	11.7
45.0	-32.0	412	13.6
90.0	-53.0	396	15.5

Maximum Energy Maneuver. The cont .s used to simulate this maneuver were zero angle of attack, maximum thrust, with wings level. The purpose of the maneuver is to gain as much airspeed as possible in a short period of time. Table IX indicates the results attained.

Table IX
Maximum Energy Maneuver

Pitch Angle-θ (deg)	Angle of Attack-α (deg)	Time (sec)	Velocity-V_T (ft/sec)
0	3.13	0.0	650
-5	0.27	3.0	669
-10	0.02	5.0	707
-15	0.00	7.0	756
-20	0.00	9.5	824
-25	0.00	11.7	887
-30	0.00	15.0	992

Maximum Rate Turn. To accomplish this maneuver, a maximum roll rate was commanded to achieve a bank angle of 135°. Then using maximum angle of attack, the nose low, slicing turn was initiated. The results indicated a very rapid turn rate once the desired bank was attained. Table X displays these results.

Table X
Maximum Rate Turn

Bank Angle-ϕ (deg)	Pitch Angle-θ (deg)	Heading Angle-ψ (deg)	Time (sec)	Velocity-V_T (ft/sec)
0	0.00	0.0	0.0	700
45	-0.14	1.2	1.6	701
90	-3.00	1.3	1.9	703
135	-49.00	41.8	5.0	687
135	-86.00	120.0	8.0	676

Defensive Maneuvers

Vertical Rolling Scissors. To generate this maneuver, the target is rolled to an inverted attitude using maximum angle of attack. As the target nears a vertical attitude, a 90° bank was commanded in an attempt to force the opponent to overshoot the target's flight path. A bank was then desired in the opposite direction to allow the target to maneuver toward the opponent's six o'clock position. Table XI shows these results versus a non-maneuvering opponent.

Table XI
Vertical Rolling Scissors

Bank Angle-ϕ (deg)	Pitch Angle-θ (deg)	Angle of Attack-α (deg)	Time (sec)	Velocity-V_T (ft/sec)
0.0	0.0	3.13	0.0	750
90.0	4.4	28.53	2.5	695
180.0	-66.0	30.00	6.5	629
90.0	-83.0	30.00	10.0	553
0.0	-18.0	30.00	14.5	478

Scissors. The scissors maneuver accomplishes a series of turn reversals in an attempt to force the opponent to the target's twelve o'clock position. These turns were performed at 90° of bank, maximum angle of attack, and idle thrust. The results of this maneuver are shown in Table XII against a non-maneuvering opponent.

Table XII
Scissors

Bank Angle-ϕ (deg)	Pitch Angle-θ (deg)	Angle of Attack-α (deg)	Time (sec)	Velocity-V_T (ft/sec)
0	0	3.13	0.0	750
-90	5.15	29.27	3.0	678
0	49.00	30.00	6.7	590
90	33.00	30.00	10.3	509

Offensive Maneuvers

Pursuit Curves. For lead, lag, and pure pursuit maneuvers, the target was programmed to null its steering error and match that error with the desired pursuit angle. By varying the desired bank angle, angle of attack, and thrust, based upon the magnitude of steering error, the target was placed upon the desired pursuit course. In testing all of the pursuit maneuvers against a non-maneuvering target, the control inputs appeared to hold the target within 5° of the desired pursuit curve.

High Speed Yo-Yo. The purpose of the high speed yo-yo is to maneuver in both the vertical and horizontal planes to preclude overshooting the opponent by possessing an excessive amount of overtake. This was accomplished by commanding a roll of 90° from the target's original bank. By rolling out of the opponent's plane, and using

maximum angle of attack and idle thrust, he was able to maintain the attack role and stay in the opponent's rear hemisphere. As the range rate increased following the rolling maneuver, the target selected a pure or lead pursuit curve to remain on the attack. The high speed yo-yo parameters versus a non-maneuvering opponent are given in Table XIII.

Table XIII
High Speed Yo-Yo

Bank Angle-ϕ (deg)	Pitch Angle-θ (deg)	Angle of Attack-α (deg)	Time (sec)	Velocity-V_T (ft/sec)
0.0	0.0	3.13	0.0	900
45.0	3.0	8.13	.7	891
90.0	5.0	12.39	3.0	875
90.0	-3.2	29.14	7.4	821
90.0	-6.7	30.00	9.3	793
45.0	-4.7	2.39	11.4	817
0.0	-6.7	2.39	15.0	884

Low Speed Yo-Yo. This offensive maneuver was utilized to increase the target's overtake while reducing angle off. To simulate the low speed yo-yo, the target was programmed to turn with the opponent aircraft. Once the angle off was less than 35°, an acceleration was commanded to increase the target's velocity. Maximum thrust, zero angle of attack, and actual bank angle were used for the acceleration. Table XIV shows the results of this maneuver.

Barrel Roll Attack. The barrel roll attack proved to be the most difficult maneuver to realistically simulate. Four sets of controls are required to simulate each segment of this maneuver. The desired control inputs are heavily dependent on relative dynamics of both

Table XIV
Low Speed Yo-Yo

Bank Angle-ϕ (deg)	Pitch Angle-θ (deg)	Angle of Attack-α (deg)	Time (sec)	Velocity-V_T (ft/sec)
0.0	0.0	3.13	0.0	800
-45.0	1.3	8.13	1.0	789
-85.0	-1.8	13.13	3.0	777
-85.0	-2.9	1.57	4.0	788
-72	-3.4	0.00	6.5	834
-72	-6.8	0.00	10.0	895

aircraft, and the barrel roll is selected for five different situation cells. For these reasons, an adequate simulation was not accomplished. The present controls included in SUBROUTINE DESIRE for the barrel roll attack are definitely inadequate, mainly due to the unrealistic method of switching between the desired controls for each maneuver segment.

Attack Maneuvers

All of the attack maneuvers utilized the desired controls from the pursuit curves. The missile attack was simulated using the pure pursuit controls. The gun attack used a lead pursuit of 20°, and the head-on gun attack was performed with a 10° lead angle. The results of the attack maneuvers showed that the simulated parameters necessary to achieve a kill may be overly restrictive. The exact pursuit curves required for these attacks were unable to be attained for the required amount of tracking time for a simulated kill.

Vita

Dennis A. Leuthauser was born on 26 August 1948 in Des Moines, Iowa where he graduated from East High School. He attended the United States Air Force Academy and graduated in June 1970 with a Bachelor of Science Degree in General Engineering. He was a football coach at the Academy in 1970 through 1971 and then completed pilot training at Williams AFB, Arizona in February 1972. He has served as a tactical fighter pilot, instructor pilot, and forward air controller at MacDill AFB, Udorn Royal Thai Air Base, and Bergstrom AFB in the F-4 and OV-10 aircraft. He reported to AFIT from Bergstrom AFB, Texas where he served as Group OV-10 Standardization/Evaluation Officer.

Permanent address: 2760 South Ocean Boulevard #·405
Palm Beach, Florida 33480

This thesis was typed by Mrs. Frances Jarnagin.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (*When Data Entered*)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
AFIT/GAE/AA/78M-7

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (*and Subtitle*)
DEVELOPMENT OF SMART TARGET FOR SIMULATION OF ONE-ON-ONE AIR-TO-AIR COMBAT

5. TYPE OF REPORT & PERIOD COVERED
MS Thesis

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
Dennis A. Leuthauser
Captain, USAF

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Air Force Institute of Technology (AFIT-EN)
Wright-Patterson AFB, Ohio 45433

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Air Force Flight Dynamics Laboratory (AFFDL-FGD)
Wright-Patterson AFB, Ohio 45433

12. REPORT DATE
March 1978

13. NUMBER OF PAGES
162

14. MONITORING AGENCY NAME & ADDRESS (*if different from Controlling Office*)

15. SECURITY CLASS. (*of this report*)
Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (*of this Report*)
Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (*of the abstract entered in Block 20, if different from Report*)

18. SUPPLEMENTARY NOTES
Approved for public release; IAW AFR 190-17

JERRAL F. GUESS, Captain, USAF
Director of Information

19. KEY WORDS (*Continue on reverse side if necessary and identify by block number*)
Air Combat Simulation
Air-to-Air Combat
Responsive Target
Air Combat Tactics
Aircraft Dynamics
Maneuvering Target
Aerial Combat
Simulation
Aircraft Simulation

20. ABSTRACT (*Continue on reverse side if necessary and identify by block number*)
Research and evaluation were conducted to determine if an adaptive maneuvering target could be utilized for air-to-air combat simulation in the Large Amplitude Multimode Aerospace Research Simulator (LAMARS). A computer program was developed, based on Quest Corporation's Smart Target, which enabled the target to make tactical decisions based on aircraft states, make variations in decision parameters corresponding to differences in pilots, and to provide control inputs to fly actual air combat maneuvers. Validation of the simulated target was accomplished by numerous test runs to ensure simulated maneuvers were realistic,

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (*When Data Entered*)

Continuation of Block 20. ABSTRACT

and continuity between maneuvers at different pilot skill levels was valid.